Package 'UKFE'

January 27, 2022

Type Package

Title UK Flood Estimation

Version 0.2.2

Author Anthony Hammond

Maintainer Anthony Hammond <agqhammond@gmail.com>

Description Functions to implement the methods of the Flood Estimation Handbook (FEH), associated updates and the revitalised flood hydrograph model (ReFH). Currently the package uses NRFA peak flow dataset version 10. Aside from FEH functionality, further hydrological functions are available. Most of the methods implemented in this package are described in one or more of the following: ``Flood Estimation Handbook'', Centre for Ecology & Hydrology (1999, ISBN:0 948540 94 X). ``Flood Estimation Handbook Supplementary Report No. 1'', Kjeldsen (2007, ISBN:0 903741 15 7). ``Regional Frequency Analysis - an approach based on L-moments'', Hosking & Wallis (1997, ISBN: 978 0 521 01940 8). ``Proposal of the extreme rank plot for extreme value analysis: with an emphasis on flood frequency studies'', Hammond (2019, <doi:10.2166/nh.2019.157>). ``Making better use of local data in flood frequency estimation'', Environment Agency (2017, ISBN: 978 1 84911 387 8). ``Sampling uncertainty of UK design flood estimation'' , Hammond (2021, <doi:10.2166/nh.2021.059>). ``Improving the FEH statistical procedures for flood frequency estimation'', Environment Agency (2008, ISBN: 978 1 84432 920 5). ``Low flow estimation in the United Kingdom'', Institute of Hydrology (1992, ISBN 0 948540 45 1). Wallingford HydroSolutions, (2016, <http://software.hydrosolutions.co.uk/winfap4/Urban-Adjustment-Procedure-Technical-Note.pdf>). Data from the UK National River Flow Archive (<https://nrfa.ceh.ac.uk/>, terms and conditions: <https://nrfa.ceh.ac.uk/costs-terms-and-conditions>).

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.2

Depends R (>= 3.5.0)

NeedsCompilation no

Repository CRAN

Date/Publication 2022-01-27 11:40:02 UTC

R topics documented:

`AMextract` *Annual maximum extraction*

Description

Extracts the annual maximum peaks from a data.frame which has dates in the first column and variable in the second.

Usage

```
AMextract(x, Plot = TRUE)
```

Arguments

x	a data.frame with dates (or POSIXct) in the first column and variable in the second
Plot	a logical argument with a default of TRUE. If TRUE the extracted annual maximum is plotted

Details

The peaks are extracted based on the UK hydrological year, which starts October 1st and ends September 30th. If there are partial years (years with missing data) the maximum value may not be the true annual maximum of the year. If there are NAs for full years in the data, an -Inf will be returned for that year.

Value

a data.frame with columns; WaterYear and AM

Author(s)

Anthony Hammond

Examples

```
#Extract the Thames AMAX daily mean flow and display the first six rows
ThamesAM <- AMextract(ThamesPQ[,c(1,3)])
head(ThamesAM)
```

AMplot *Plot of the annual maximum sample*

Description

Provides two plots. First, a histogram of the sample, second, a barplot

Usage

```
AMplot(x)
```

Arguments

x	a data.frame with at least two columns. The first a date column and the second the annual maximum (AM) sequence. A third column with the station id is necessary for inclusion of the id in the plot title. An AM object derived from the GetAM or ImportAM functions.

Details

When used with a GetAM object or any data.frame with dates in the first column, the barplot is daily. Therefore, although it's an annual maximum (AM) sequence, some bars will be closer together depending on the number of days between them.

Value

a histogram of the AMAX sample and a barplot

Author(s)

Anthony Hammond

Examples

```
#Get an AMAX sample and plot
AMplot(GetAM(58002))
```

AMSP *National River Flow Archive (NRFA) annual maximum data for sites suitable for pooling*

Description

A data.frame with three columns; Date, Flow, id. NRFA Peak Flow Dataset - Version 10

Usage

```
AMSP
```

Format

A data frame with 25020 rows and 3 columns

Date Date

Flow Annual maximum peak flow, m3/s

id Station identification number

Source

https://nrfa.ceh.ac.uk/peak-flow-dataset

ARF *Areal reduction factor (ARF)*

Description

The results of applying, to a rainfall depth, the ratio of the rainfall over an area to the rainfall depth of the same duration at a representative point in the area.

Usage

```
ARF(Depth, Area, D)
```

Arguments

Depth	depth of rainfall
Area	catchment area in km2
D	duration in hours

Details

The ARF and it's use is detailed in the Flood Estimation Handbook (1999), volume 2. The DDF model is calibrated on point rainfall and the areal reduction factor converts it to a catchment rainfall for use with a rainfall runoff model such as ReFH (see details for ReFH function). The ReFH model includes a design rainfall profile for winter and summer but the depth duration frequency (DDF) model is calibrated on annual maximum peaks as opposed to seasonal peaks. A seasonal correction factor (SCF) is necessary to convert the DDF estimate to a seasonal one. The final depth, therefore is; Depth = DDFdepth x ARF x SCF.

Value

the rainfall depth or rainfall return period

Author(s)

Anthony Hammond

Examples

```
#Derive the ARF for a depth of 30, an area of 500km2 and a duration of 12 hours
ARF(30, 500, 12)
```

`BFI` *Baseflow index (BFI)*

Description

Calculates the baseflow index from a daily mean flow series

Usage

```
BFI(Q, x.lim = NULL, y.lim = NULL, PlotTitle = "Baseflow plot", Plot = TRUE)
```

Arguments

`Q`	the daily mean flow series. Numeric vector
`x.lim`	the x axis limits of the plot. Numeric vector of length two Default is the extents of the data
`y.lim`	the y axis limits of the plot. Numeric vector of length two. Default is the extents of the data
`PlotTitle`	the title of the plot. The default is "Baseflow plot"
`Plot`	a logical argument with a default of TRUE. If TRUE the daily flow is plotted with the baseflow highlighted.

Details

The baseflow index is calculated using the method outlined in Gustard, A. Bullock, A. Dixon, J. M.. (1992). Low flow estimation in the United Kingdom. Wallingford, Institute of Hydrology, 88pp. (IH Report No.108)

Value

the baseflow index and if Plot equals TRUE, a plot showing the flow time series (black) and the associated baseflow (red)

Author(s)

Anthony Hammond

Examples

```
# Calculate the BFI from daily discharge at Kingston upon Thames;
# which is in column three of the ThamesPQ data
BFI(ThamesPQ[,3])
```

DDF99 *FEH99 depth duration frequency precipitation model*

Description

Estimation of design rainfall depths, and the rarity of observed rainfall

Usage

```
DDF99(D, RP, pars, Depth = NULL, disc = NULL)
```

Arguments

`D`	numeric. The duration of interest (in hours)
`RP`	return period
`pars`	a numeric vector of length six. The six catchment parameters for the DDF model in the order of: c, d1, d2, d3, e, f
`Depth`	a user supplied rainfall depth for the duration under question
`disc`	converts from the sliding duration to fixed duration estimate. Choices are "hourly" or "daily"

Details

The depth duration frequency rainfall model is detailed in the Flood Estimation Handbook (1999), volume 2. A note about the discretisation: The user can choose between "daily" or "hourly" for the sliding duration to fixed duration conversion. If the 'Depth' argument is used, it overrides the return period (RP) argument and provides RP as a function of depth. However, if both the 'Depth' and the 'disc' arguments are used, the sliding duration depth is provided as a function of the user input depth. This resulting depth can then be used without the 'disc' argument to determine the sliding duration RP.

Value

the rainfall depth or rainfall return period

Author(s)

Anthony Hammond

Examples

```
#Examples from FEH volume 2
#The parameters for these examples are from FEH v2
#What is the 2-day rainfall with return period 100-years for Norwich.
DDF99(D = 48, RP = 100, pars = c(-0.023, 0.273, 0.351, 0.236, 0.309, 2.488))
#What is the 4-hour rainfall with return period 20 years for a typical point in the Lyne catchment
DDF99(D = 4, RP = 20, pars = c(-0.025, 0.344, 0.485, 0.402, 0.287, 2.374))
```

```
#How rare was the rainfall of 6th August 1978 at Broughshane, County Antrim?
DDF99(D = 5, Depth = 47.7, pars = c(-0.022, 0.412, 0.551, 0.276, 0.261, 2.252))
```

DesHydro *Design hydrograph extraction*

Description

Extracts a mean hydrograph from a flow series

Usage

```
DesHydro(
  x,
  qu = 0.8,
  n = 10,
  thr = 0.975,
  xst = NULL,
  xend = NULL,
  RetAll = FALSE,
  Smooth = 1
)
```

Arguments

`x`	a numeric vector. The flow series of interest
`qu`	the quantile of flow which separates peaks and truncates either side of the peak to form the event hydrograph. The default is 0.8
`n`	number of event hydrographs from which to derive the mean hydrograph. Default is 10. Depending on the length of x, there may be less than 10
`thr`	threshold above which event peaks are selected. The default is 0.975
`xst`	an integer to truncate the x axis of the plot and resulting design hydrograph. The first point of the design hydrograph
`xend`	an integer to truncate the x axis of the plot and resulting design hydrograph. The last point of the design hydrograph
`RetAll`	logical argument with a default of false. If TRUE, all the hydrographs from which the mean is derived are returned in a data.frame. If FALSE, the mean hydrograph is returned
`Smooth`	an integer (from 0 to 5). To smooth the design hydrograph. The default is 1 which provides the minimum level of smoothing. 0 is no smoothing and 5 is the highest

Details

All the peaks over a user defined threshold are identified and separated by a user defined value 'qu', which is a quantile of flow. The top n peaks are selected and the hydrographs extracted. Each hydrograph is centred on the peak and truncated either side, where the flow falls below the 'qu' quantile flow. All events are scaled to have a peak flow of one, and the mean of these is taken as the scaled design hydrograph. After an initial view of the hydrograph, it can be truncated using the 'xst' and 'xend' arguments. The default is to select 10 hydrographs for averaging, however, there may well be fewer if the sample is short.

Value

a numeric vector which is the mean of the top n peak events in the flow series. Also a plot of the n hydrographs and the design hydrograph. If the RetAll argument equals TRUE, a data.frame of the n hydrographs is returned instead.

Note

The smoothing is done by rolling average, where the the mean is of points from n to the left up to n to the right. The n is chosen by the Smooth argument.

Author(s)

Anthony Hammond

Examples

```
#Extract a design hydrograph from the Thames daily mean flow. Then print the resulting hydrograph
ThamesDesHydro <- DesHydro(ThamesPQ$Q)
ThamesDesHydro
#Do the same but truncate the design hydrograph and the plot from the first point to the 30th
DesHydro(ThamesPQ$Q, xst = 1, xend = 30)
#adjust the qu value to see the impact
ThamesDesHydro <- DesHydro(ThamesPQ$Q, qu = 0.7)
#Return all the hydrographs
ThamesHydros <- DesHydro(ThamesPQ$Q, xst = 1, xend = 30, RetAll = TRUE)
#view the first six rows of the hydrographs
head(ThamesHydros)
```

DeTrend *Linearly detrend a sample*

Description

Applies a linear detrend to a sample

Usage

```
DeTrend(x)
```

Arguments

x	a numeric vector

Details

Adjusts all the values in the sample, of size n, by the difference between the linearly modelled ith data point and the linearly modelled nth data point.

Value

A numeric vector which is a linearly detrended version of x.

Author(s)

Anthony Hammond

Examples

```
# Get an annual maximum (AM) sample that looks to have a significant trend
AM.21025 <- GetAM(21025)
# plot the resulting AM as a bar plot. Then detrend and compare with another plot
plot(AM.21025$Flow, type = "h", ylab = "Discharge (m3/s)")
AM.Detrend <- DeTrend(AM.21025$Flow)
plot(AM.Detrend, type = "h", ylab = "Discharge (m3/s)")
```

`DiagPlots` *Diagnostic plots for pooling groups*

Description

Provides 10 plots to compare the sites in the pooling group

Usage

```
DiagPlots(x, gauged = FALSE)
```

Arguments

x	pooling group derived from the Pool() function
gauged	logical argument with a default of FALSE. TRUE adds the top site in the pooling group to the plots in a different colour

Value

ten diagnostic plots for pooling groups

Author(s)

Anthony Hammond

Examples

```
#Form a gauged pooling group and plot the diagnostics with gauged = TRUE
Pool.96001 <- Pool(GetCDs(96001))
DiagPlots(Pool.96001, gauged = TRUE)
#Form an ugauged pooling group and plot the diagnostics
Pool.96001 <- Pool(GetCDs(96001), exclude = 96001)
DiagPlots(Pool.96001)
```

DonAdj *Donor adjustment candidates & results*

Description

Provides donor adjustment candidates, descriptors, and results in order of the proximity to the centroid of the subject catchment.

Usage

```
DonAdj(CDs = NULL, x, y, QMEDscd = NULL, alpha = TRUE, rows = 10, d2 = NULL)
```

Arguments

`CDs`	catchment descriptors derived from either GetCDs or ImportCDs
`x`	catchment centroid easting (for when CDs isn't used)
`y`	catchment centroid northing (for when CDs isn't used)
`QMEDscd`	QMED estimate for the catchment of interest (for when CDs isn't used)
`alpha`	logical argument with a default of TRUE. If FALSE the exponent of the donor adjustment equation is set to one
`rows`	number of sites provided; default is 10
`d2`	a numeric vector of length two; the two site references for the donor catchments chosen for the two donor case

Details

When d2 is FALSE the results for single donor adjustment are in the final column headed 'QMED.adj' for each site. If alpha is set to FALSE, the results in this column are from the same donor equation but with an exponent of 1. The donor adjustment method is as outlined in Science Report: SC050050 - Improving the FEH statistical procedures for flood frequency estimation. The method for two donors is outlined in 'Kjeldsen, T. (2019). Adjustment of QMED in ungauged catchments using two donor sites. Circulation - The Newsletter of the British Hydrological Society, 4'. When two donors are used, only the result is returned, rather than donor candidates. The QMEDfse column provides the gauged factorial standard error for the median of the annual maximum sample. It is worth considering this when choosing a donor site (a high value indicates a poor donor). When choosing between two donors, the site with a lower QMEDfse would be an appropriate choice (all else being equal). The QMEDfse is calculated with the QMEDfseSS() function.

Value

A data.frame with rownames of site references and columns of catchment descriptors, distance from subect site, and associated results. When two donors are used, only the resulting adjusted QMED is returned

Author(s)

Anthony Hammond

Examples

```
#Get some CDs and output candidate donor sites
CDs.54022 <- GetCDs(54022)
DonAdj(CDs.54022)
#Get results with inputs of x,y, and QMEDscd
DonAdj(x = 283261, y = 288067, QMEDscd = 17.931)
#Get a result with two donors
DonAdj(CDs.54022, d2 = c(54092, 54091))
```

EncProb *Encounter probabilities*

Description

Calculates the probability of experiencing at least n events with a given return period (RP), over a given number of years

Usage

```
EncProb(n, yrs, RP, dist = "Poisson")
```

Arguments

n	number of events
yrs	number of years
RP	return period of the events
dist	choice of probability distribution. Either "Poisson" or "Binomial"

Details

The choice of binomial or Poisson distributions for calculating encounter probablities is akin to annual maximum (AM) versus peaks over threshold (POT) approaches to extreme value analysis. AM and binomial assume only one "event" can occur in the blocked time period. Whereas Poisson and POT don't make this assumption. In the case of most catchments in the UK, it is rare to have less than two independent "events" per year; in which case the Poisson and POT choices are more suitable. In large catchments, with seasonally distinctive baseflow, there may only be one independent peak in the year. However, the results from both methods converge with increasing magnitude, yielding insignificant difference beyond a 20-year return period.

Value

A probability

Author(s)

Anthony Hammond

Examples

```
#Calculate the probability of exceeding at least one 50-yr RP event
#over a 10 year period, using the Poisson distribution.
EncProb(n = 1, yrs = 10, RP = 50)
#Calculate the probability of exceeding at least two 100-yr RP events
#over a 100 year period, using the binomial distribution.
EncProb(n = 2, yrs = 100, RP = 100, dist = "Binomial")
```

ERPlot *Extreme rank plot*

Description

A plot to inspect the distribution of ordered data

Usage

```
ERPlot(
  x,
  Title = "Extreme Rank Plot",
  dist = "GenLog",
  pars = NULL,
  GF = NULL,
  ylabel = "Discharge (m3/s)",
  ln = FALSE
)
```

Arguments

x	numeric vector. A sample for inspection
Title	a character string to change the default title, which is "Extreme Rank Plot"
dist	a choice of distribution. The choices are "GenLog", "GEV", "Gumbel" and "GenPareto"
pars	a vector of length three. In the order of location, scale, & shape. If left null the parameters are estimated from x
GF	a vector of length three, in the order of; Lcv, LSkew and Median
ylabel	a character string. For user choice of a label for the y axis.
ln	logical TRUE or FALSE with a default of FALSE. If TRUE, the variable under consideration is log transformed for the plot

Details

By default the parameters of the distribution for comparison with the sample are estimated from the sample. However, the pars argument can be used to compare the distribution with parameters estimated separately. Similarly the growth factor (GF) parameters, linear coefficient of variation (Lcv) & linear skewness (LSkew) with the median can be entered. In this way the pooling estimated distribution can be compared to the sample. The ERplot is described in Hammond, A. (2019). Proposal of the 'extreme rank plot' for extreme value analysis: with an emphasis on flood frequency studies. Hydrology Research, 50 (6), 1495–1507.

Value

The extreme rank plot with GoTF scores

Author(s)

Anthony Hammond

Examples

```
#Get an AMAX sample and plot
AM.27083 <- GetAM(27083)
ERPlot(AM.27083$Flow)
#Get some pooled estimate of Lcv & LSkew to use with the GF argument
QuickResults(GetCDs(27083), gauged = TRUE)
#Use the resulting Lcv, Lskew and RP2 for the GF argument and change the title
ERPlot(AM.27083$Flow, Title = "Site 27083 pooled comparison", GF = c(0.2286109, 0.1536903, 12.513))
```

EVPlot *Extreme value plot (frequency and growth curves)*

Description

Plots the extreme value frequency curve or growth curve with observed sample points.

Usage

```
EVPlot(
  x,
  dist = "GenLog",
  scaled = TRUE,
  Title = "Extreme value plot",
  ylabel = NULL,
  LineName = NULL,
  Unc = TRUE
)
```

Arguments

x	a numeric vector. The sample of interest
dist	a choice of distribution. "GEV", "GenLog", "Gumbel" or "GenPareto". The default is "GenLog"
scaled	logical argument with a default of TRUE. If TRUE the plot is a growth curve (scaled by the QMED). If FALSE, the plot is a frequency curve
Title	a character string. The user chosen plot title. The default is "Extreme value plot"
ylabel	a character string. The user chosen label for the y axis. The default is "Q/QMED" if scaled = TRUE and "Discharge (m3/s)" if scaled = FALSE
LineName	a character string. User chosen label for the plotted curve
Unc	logical argument with a default of TRUE. If TRUE, 95 percent uncertainty intervals are plotted.

Details

The plotting has the option of generalised extreme value (GEV), generalised Pareto (GenPareto), Gumbel, or generalised logistic (GenLog) distributions. The uncertainty is quantified by bootstrapping.

Value

An extreme value plot (freqency or growth curve) with intervals to quantify uncertainty

Author(s)

Anthony Hammond

Examples

```
#Get an AMAX sample and plot the growth curve with the GEV distribution
AM.203018 <- GetAM(203018)
EVPlot(AM.203018$Flow, dist = "GEV")
```

EVPlotAdd *Add lines and/or points to an extreme value plot*

Description

Functionality to add extra lines or points to an extreme value plot (derived from the EVPlot function).

Usage

```
EVPlotAdd(
  Pars,
  dist = "GenLog",
  Name = "Adjusted",
  MED = NULL,
  xyleg = NULL,
  col = "red",
  lty = 1,
  pts = NULL,
  ptSym = NULL
)
```

Arguments

`Pars`	a numeric vector of length two. The first is the Lcv (linear coefficient of variation) and the second is the Lskew (linear skewness).
`dist`	distribution name with a choice of "GenLog", "GEV", "GenPareto", and "Gumbel"
`Name`	character string. User chosen name for points or line added (for the legend)
`MED`	The two year return level. Necessary In the case where the EV plot is not scaled
`xyleg`	a numeric vector of length two. They are the x and y position of the symbol and text to be added to the legend.
`col`	The colour of the points of line that have been added
`lty`	An integer. The type of line added
`pts`	A numeric vector. An annual maximum sample, for example. This is for points to be added
`ptSym`	An integer. The symbol of the points to be added

Details

A line can be added using the Lcv and Lskew based on one of four distributions (Generalised extreme value, Generalised logistic, Gumbel, Generalised Pareto). Points can be added as a numeric vector. If a single point is required, the base points() function can be used and the x axis will need to be log(RP-1).

Value

Additional, user specified line or points to an extreme value plot derived from the EVPlot function.

Author(s)

Anthony Hammond

Examples

```
#Get an AMAX sample and plot the growth curve with the GEV distribution
AM.203018 <- GetAM(203018)
EVPlot(AM.203018$Flow, dist = "GEV")
#Now add a line (dotted & red) for the generalised logistic distribution
#first get the Lcv and Lskew using the Lmoms function
pars <- as.numeric(Lmoms(AM.203018[,2])[c(5,6)])
EVPlotAdd(Pars = pars, dist = "GenLog", Name = "GenLog", xyleg = c(-5.2,2.65), lty = 3)
#Now add a line for the gumbel distribution which is darkgreen and dashed.
EVPlotAdd(Pars = pars[1], dist = "Gumbel", Name = "Gumbel",
xyleg = c(-5.19,2.5), lty = 3, col = "darkgreen")
#now plot afresh and get another AMAX and add the points
EVPlot(AM.203018$Flow, dist = "GEV")
AM.27090 <- GetAM(27090)
EVPlotAdd(xyleg = c(-4.9,2.65), pts = AM.27090[,2], Name = "27090")
```

`EVPool` *Extreme value plot for pooling groups*

Description

Plots the extreme value frequency curve or growth curve for gauged or ungauged pooled groups

Usage

```
EVPool(
  x,
  AMAX = NULL,
  gauged = FALSE,
  dist = "GenLog",
  QMED = NULL,
  Title = "Pooled growth curve",
  UrbAdj = FALSE,
  CDs
)
```

Arguments

x	pooling group derived from the Pool() function
AMAX	the AMAX sample to be plotted in the case of gauged. If NULL, & gauged equals TRUE, the AMAX from the first site in the pooling group is used
gauged	logical argument with a default of FALSE. If FALSE, the plot is the ungauged pooled curve accompanied by the single site curves of the group members. If TRUE, the plot is the gauged curve and single site curve with the observed points added
dist	a choice of distribution. Choices are "GEV" or "GenLog". The default is "GenLog"

`QMED`	a chosen QMED to convert the curve from a growth curve to the frequency curve
`Title`	a character string. The user chosen plot title. The default is "Pooled growth curve"
`UrbAdj`	a logical argument with a default of FALSE. If TRUE and urban adjustment is applied to the pooled growth curve
`CDs`	catchment descriptors derived from either GetCDs or ImportCDs. Only necessary if UrbAdj is TRUE

Value

An extreme value plot for gauged or ungauged pooling groups

Author(s)

Anthony Hammond

Examples

```
#Get some CDs, form an ungauged pooling group and apply EVPlot.
CDs.96001 <- GetCDs(96001)
Pool.96001 <- Pool(CDs.96001, exclude = 96001)
EVPool(Pool.96001)
#Do the same for the gauged case, change the title, and convert with a QMED of 105.5.
PoolG.96001 <- Pool(CDs.96001)
EVPool(PoolG.96001, gauged = TRUE, Title = "Gauged frequency curve - Site 96001", QMED = 105.5)
#Pretend we have an extra AMAX for the gauge. Amend the pooling group Lcv and LSkew
#for the site accordingly then apply EVPool with the updated AMAX.
#Firstly, get the AMAX sample
AM.96001 <- GetAM(96001)
#Add an extra AMAX flow of 350m3/s
Append96001 <- append(AM.96001$Flow, 350)
#Amend the Lcv and Lskew in the pooling group
PoolG.96001[1, c(16, 17)] <- c(Lcv(Append96001), LSkew(Append96001))
#Now plot gauged with the updated AMAX
EVPool(PoolG.96001, AMAX = Append96001, gauged = TRUE)
```

`GenLogAM` *Generalised logistic distribution - estimates directly from sample*

Description

Estimated quantiles as a function of return period (RP) and vice versa, directly from the data

Usage

```
GenLogAM(x, RP = 100, q = NULL, trend = FALSE)
```

Arguments

x	numeric vector (block maxima sample)
RP	return period (default = 100)
q	quantile (magnitude of variable)
trend	logical argument with default of FALSE. If TRUE, a linear adjustment to the location parameter is made to account for non-stationarity

Details

If the argument q is used, it overrides RP and provides RP as a function of q (magnitude of variable) as opposed to q as a function of RP. The parameters are estimated by the method of L-moments, as detailed in 'Hosking J. Wallis J. 1997 Regional Frequency Analysis: An Approach Based on L-moments. Cambridge University Press, New York'. The trend argument allows the location parameter to move in line with the observed linear trend of the sample. Another option is to detrend the sample first with the DeTrend function. On average this makes little difference to the two year flow but lower results for longer return periods (not always) when compared to the trend option in this function.

Value

quantile as a function of RP or vice versa, with the option of accounting for the linear trend in the sample

Author(s)

Anthony Hammond

Examples

```
#Get an annual maximum sample and estimate the 50-year RP
AM.27090 <- GetAM(27090)
GenLogAM(AM.27090$Flow, RP = 50)
#Estimate the RP for a 600m3/s discharge
GenLogAM(AM.27090$Flow, q = 600)
#Estimate the 50-year RP allowing for non-stationarity in the location parameter
GenLogAM(AM.27090$Flow, RP = 50, trend = TRUE)
```

GenLogEst *Generalised logistic distribution estimates from parameters*

Description

Estimated quantiles as function of return period (RP) and vice versa, from user input parameters

Usage

```
GenLogEst(loc, scale, shape, q = NULL, RP = 100)
```

Arguments

loc	location parameter
scale	scale parameter
shape	shape parameter
q	quantile. magnitude of the variable under consideration
RP	return period

Details

If the argument q is used, it overrides RP and provides RP as a function of q (magnitude of variable) as opposed to q as a function of RP.

Value

quantile as a function of RP or vice versa

Author(s)

Anthony Hammond

Examples

```
#Get an annual maximum sample, estimate the parameters and estimate 50-year RP
AM.27090 <- GetAM(27090)
GenLogPars(AM.27090$Flow)
#Store parameters in an object
Pars <- as.numeric(GenLogPars(AM.27090$Flow))
#get estimate of 50-yr flow
GenLogEst(Pars[1], Pars[2], Pars[3], RP = 50)
#Estimate the RP for a 600m3/s discharge
GenLogEst(Pars[1], Pars[2], Pars[3], q = 600)
```

GenLogGF *Generalised logistic distribution growth factors*

Description

Estimated growth factors as a function of return period, with inputs of Lcv & LSkew (linear coefficient of variation & linear skewness)

Usage

```
GenLogGF(lcv, lskew, RP)
```

Arguments

lcv	linear coefficient of variation
lskew	linear skewness
RP	return period

Details

Growth factors are calculated by the method outlined in the Flood Estimation Handbook, volume 3, 1999.

Value

Generalised logistic estimated growth factor

Author(s)

Anthony Hammond

Examples

```
#Estimate the 50-year growth factors from an Lcv and Lskew of 0.17 and 0.04, respectively.
GenLogGF(0.17, 0.04, RP = 50)
```

GenLogPars *Generalised logistic distribution parameter estimates*

Description

Estimated parameters from a sample (with Lmoments or maximum likelihood estimation) or from L1 (first L-moment), Lcv (linear coefficient of variation), and LSkew (linear skewness)

Usage

```
GenLogPars(x = NULL, mle = FALSE, L1, LCV, LSKEW)
```

Arguments

x	numeric vector. The sample
mle	logical argument with a default of FALSE. If FALSE the parameters are estimated with Lmoments, if TRUE the parameters are estimated by maximum likelihood estimation.
L1	first Lmoment
LCV	linear coefficient of variation
LSKEW	linear skewness

Details

The L-moment estimated parameters are by the method detailed in 'Hosking J. Wallis J. 1997 Regional Frequency Analysis: An Approach Based on L-moments. Cambridge University Press, New York'

Value

Parameter estimates (location, scale, shape)

Author(s)

Anthony Hammond

Examples

```
#Get an annual maximum sample and estimate the parameters using Lmoments
AM.27090 <- GetAM(27090)
GenLogPars(AM.27090$Flow)
#Estimate parameters using MLE
GenLogPars(AM.27090$Flow, mle = TRUE)
#calculate Lmoments and estimate the parmeters with L1, Lcv and Lskew
Lmoms(AM.27090$Flow)
#store linear moments in an object
LPars <- as.numeric(Lmoms(AM.27090$Flow))[c(1,5,6)]
GenLogPars(L1 = LPars[1], LCV = LPars[2], LSKEW = LPars[3])
```

GenParetoEst — *Generalised Pareto distribution estimates from parameters*

Description

Estimated quantiles as function of return period (RP) and vice versa, from user input parameters

Usage

```
GenParetoEst(loc, scale, shape, q = NULL, RP = 100, ppy = 1)
```

Arguments

loc	location parameter
scale	scale parameter
shape	shape parameter
q	quantile. magnitude of the variable under consideration
RP	return period
ppy	peaks per year. Default is one

Details

If the argument q is used, it overrides RP and provides RP as a function of q (magnitude of variable) as opposed to q as a function of RP. The average number of peaks per year argument (ppy) is for the function to convert from the peaks over threshold (POT) scale to the annual scale. For example, if there are 3 peaks per year, the probability associated with the 100-yr return period estimate would be 0.01/3 (i.e. an RP of 300 on the POT scale) rather than 0.01.

Value

quantile as a function of RP or vice versa

Author(s)

Anthony Hammond

Examples

```
#Get a POT sample, estimate the parameters, and estimate 50-year RP
ThamesPOT <- POTextract(ThamesPQ[,c(1,3)], thresh = 0.90)
GenParetoPars(ThamesPOT$peak)
#Store parameters in an object
Pars <- as.numeric(GenParetoPars(ThamesPOT$peak))
#get estimate of 50-yr flow
GenParetoEst(Pars[1], Pars[2], Pars[3], ppy = 1.867, RP = 50)
#Estimate the RP for a 600m3/s discharge
GenParetoEst(Pars[1], Pars[2], Pars[3], ppy = 1.867, q = 600)
```

GenParetoGF *Generalised Pareto distribution growth factors*

Description

Estimated growth factors as a function of return period, with inputs of Lcv & LSkew (linear coefficient of variation & linear skewness)

Usage

```
GenParetoGF(lcv, lskew, RP, ppy = 1)
```

Arguments

`lcv`	linear coefficient of variation
`lskew`	linear skewness
`RP`	return period
`ppy`	peaks per year

Details

Growth factors (GF) are calculated by the method outlined in the Flood Estimation Handbook, volume 3, 1999. The average number of peaks per year argument (ppy) is for the function to convert from the peaks over threshold (POT) scale to the annual scale. For example, if there are 3 peaks per year, the probability associated with the 100-yr return period estimate would be 0.01/3 (i.e. an RP of 300 on the POT scale) rather than 0.01.

Value

Generalised Pareto estimated growth factor

Author(s)

Anthony Hammond

Examples

```
#Get POT flow data from the Thames at Kingston (noting the no. peaks per year).
#Then estimate the 100-year growth factor with lcv and lskew estimates
TPOT <- POTextract(ThamesPQ[,c(1,3)], thresh = 0.90)
GenParetoGF(Lcv(TPOT$peak), LSkew(TPOT$peak), RP = 100, ppy = 1.867)
#multiply by the median of the POT data for an estimate of the 100-yr flood
GenParetoGF(Lcv(TPOT$peak), LSkew(TPOT$peak), RP = 100, ppy = 1.867)*median(TPOT$peak)
```

`GenParetoPars` *Generalised Pareto distribution parameter estimates*

Description

Estimated parameters from a sample (with Lmoments or maximum likelihood estimation) or from L1 (first L-moment), Lcv (linear coefficient of variation), and LSkew (linear skewness)

Usage

```
GenParetoPars(x = NULL, mle = FALSE, L1, LCV, LSKEW)
```

Arguments

`x`	numeric vector. The sample
`mle`	logical argument with a default of FALSE. If FALSE the parameters are estimated with Lmoments, if TRUE the parameters are estimated by maximum likelihood estimation
`L1`	first Lmoment
`LCV`	linear coefficient of variation
`LSKEW`	linear skewness

Details

The L-moment estimated parameters are by the method detailed in 'Hosking J. Wallis J. 1997 Regional Frequency Analysis: An Approach Based on L-moments. Cambridge University Press, New York'

Value

Parameter estimates (location, scale, shape)

Author(s)

Anthony Hammond

Examples

```
#Get a peaks over threshold sample and estimate the parameters using Lmoments
ThamesPOT <- ThamesPOT <- POTextract(ThamesPQ[,c(1,3)], thresh = 0.90)
GenParetoPars(ThamesPOT$peak)
#Estimate parameters using MLE
GenParetoPars(ThamesPOT$peak, mle = TRUE)
#calculate Lmoments and estimate the parmeters with L1, Lcv and Lskew
Lmoms(ThamesPOT$peak)
#store linear moments in an object
LPars <- as.numeric(Lmoms(ThamesPOT$peak))[c(1,5,6)]
GenParetoPars(L1 = LPars[1], LCV = LPars[2], LSKEW = LPars[3])
```

GenParetoPOT — *Generalised Pareto distribution - estimates directly from sample*

Description

Estimated quantiles as function of return period (RP) and vice versa, directly from the data

Usage

```
GenParetoPOT(x, ppy = 1, RP = 100, q = NULL)
```

Arguments

x	numeric vector (block maxima sample)
ppy	peaks per year
RP	return period (default = 100)
q	quantile (magnitude of variable)

Details

If the argument q is used, it overrides RP and provides RP as a function of q (magnitude of variable) as opposed to q as a function of RP. The average number of peaks per year argument (ppy) is for the function to convert from the peaks over threshold (POT) scale to the annual scale. For example, if there are 3 peaks per year, the probability associated with the 100-yr return period estimate would be 0.01/3 (i.e. an RP of 300 on the POT scale) rather than 0.01. The parameters are estimated by the method of L-moments, as detailed in 'Hosking J. Wallis J. 1997 Regional Frequency Analysis: An Approach Based on L-moments. Cambridge University Press, New York'.

Value

quantile as a function of RP or vice versa

Author(s)

Anthony Hammond

Examples

```
#Get a POT series and estimate the 50-year RP
ThamesPOT <- POTextract(ThamesPQ[,c(1,3)], thresh = 0.90)
GenParetoPOT(ThamesPOT$peak, ppy = 1.867, RP = 50)
#Estimate the RP for a 600m3/s discharge
GenParetoPOT(ThamesPOT$peak, ppy = 1.867, q = 600)
```

GetAM   *Get an annual maximum sample from the National River Flow Archive sites suitable for pooling*

Description

Extracts the annual maximum peak flow sample and associated dates for the site of interest.

Usage

```
GetAM(ref)
```

Arguments

`ref`	the site reference of interest (numeric)

Value

A data.frame with columns; Date, Flow, and id

Author(s)

Anthony Hammond

Examples

```
#Get an AMAX sample and display it in the console
GetAM(203018)
#Save an AMAX sample as an object
AM.203018 <- GetAM(203018)
```

GetCDs *Get catchment descriptors from the National River Flow Archive sites considered suitable for median annual maximum flow estimation (QMED).*

Description

Extracts the catchment descriptors for a site of interest from those suitable for QMED.

Usage

```
GetCDs(x)
```

Arguments

x	the site reference of interest (numeric)

Value

A data.frame with columns; Descriptor and Value.

Author(s)

Anthony Hammond

Examples

```
#Get CDs and display in the console
CDs.203018 <- GetCDs(203018)
CDs.203018
```

`GetQMED` *QMED from a gauged site suitable for QMED*

Description

Provides QMED (median annual maximum flow) from a site suitable for QMED, using the site reference.

Usage

```
GetQMED(x)
```

Arguments

`x`	the gauged reference

Value

the median annual maximum

Author(s)

Anthony Hammond

Examples

```
#Get the observed QMED from sites 55002
GetQMED(55002)
```

`GEVAM` *Generalised extreme value distribution - estimates directly from sample*

Description

Estimated quantiles as function of return period (RP) and vice versa, directly from the data

Usage

```
GEVAM(x, RP = 100, q = NULL, trend = FALSE)
```

Arguments

`x`	numeric vector (block maxima sample)
`RP`	return period (default = 100)
`q`	quantile (magnitude of variable)
`trend`	logical argument with default of FALSE. If TRUE, a linear adjustment to the location parameter is made to account for non-stationarity

Details

If the argument q is used, it overrides RP and provides RP as a function of q (magnitude of variable) as opposed to q as a function of RP. The parameters are estimated by the method of L-moments, as detailed in 'Hosking J. Wallis J. 1997 Regional Frequency Analysis: An Approach Based on L-moments. Cambridge University Press, New York'. The trend argument allows the location parameter to move in line with the observed linear trend of the sample. Another option is to detrend the sample first with the DeTrend function. On average this makes little difference to the two year flow but lower results for longer return periods (not always) when compared to the trend option in this function.

Value

quantile as a function of RP or vice versa, with the option of accounting for the linear trend in the sample

Author(s)

Anthony Hammond

Examples

```
#Get an annual maximum sample and estimate the 50-year RP
AM.27090 <- GetAM(27090)
GEVAM(AM.27090$Flow, RP = 50)
#Estimate the RP for a 600m3/s discharge
GEVAM(AM.27090$Flow, q = 600)
#Estimate the 50-year RP allowing for non-stationarity in the location parameter
GEVAM(AM.27090$Flow, RP = 50, trend = TRUE)
```

GEVEst — *Generalised extreme value distribution estimates from parameters*

Description

Estimated quantiles as function of return period (RP) and vice versa, from user input parameters

Usage

```
GEVEst(loc, scale, shape, q = NULL, RP = 100)
```

Arguments

loc	location parameter
scale	scale parameter
shape	shape parameter
q	quantile. magnitude of the variable under consideration
RP	return period

Details

If the argument q is used, it overrides RP and provides RP as a function of q (magnitude of variable) as opposed to q as a function of RP.

Value

quantile as a function of RP or vice versa

Author(s)

Anthony Hammond

Examples

```
#Get an annual maximum sample, estimate the parameters and estimate 50-year RP
AM.27090 <- GetAM(27090)
GEVPars(AM.27090$Flow)
#Store parameters in an object
Pars <- as.numeric(GEVPars(AM.27090$Flow))
#get estimate of 50-yr flow
GEVEst(Pars[1], Pars[2], Pars[3], RP = 50)
#Estimate the RP for a 600m3/s discharge
GEVEst(Pars[1], Pars[2], Pars[3], q = 600)
```

GEVGF *Generalised extreme value distribution growth factors*

Description

Estimated growth factors as a function of return period, with inputs of Lcv & LSkew (linear coefficient of variation & linear skewness)

Usage

```
GEVGF(lcv, lskew, RP)
```

Arguments

`lcv`	linear coefficient of variation
`lskew`	linear skewness
`RP`	return period

Details

Growth factors are calculated by the method outlined in the Flood Estimation Handbook, volume 3, 1999.

Value

Generalised extreme value estimated growth factor

Author(s)

Anthony Hammond

Examples

```
#Estimate the 50-year growth factors from Lcv = 0.17 and Lskew = 0.04
GEVGF(0.17, 0.04, RP = 50)
```

GEVPars *Generalised extreme value distribution parameter estimates*

Description

Estimated parameters from a sample (with Lmoments or maximum likelihood estimation) or from L1 (first L-moment), Lcv (linear coefficient of variation), and LSkew (linear skewness)

Usage

```
GEVPars(x = NULL, mle = FALSE, L1, LCV, LSKEW)
```

Arguments

`x`	numeric vector. The sample
`mle`	logical argument with a default of FALSE. If FALSE the parameters are estimated with Lmoments, if TRUE the parameters are estimated by maximum likelihood estimation
`L1`	first Lmoment
`LCV`	linear coefficient of variation
`LSKEW`	linear skewness

Details

The L-moment estimated parameters are by the method detailed in 'Hosking J. Wallis J. 1997 Regional Frequency Analysis: An Approach Based on L-moments. Cambridge University Press, New York'

Value

Parameter estimates (location, scale, shape)

Author(s)

Anthony Hammond

Examples

```
#Get an annual maximum sample and estimate the parameters using Lmoments
AM.27090 <- GetAM(27090)
GEVPars(AM.27090$Flow)
#Estimate parameters using MLE
GEVPars(AM.27090$Flow, mle = TRUE)
#calculate Lmoments and estimate the parmeters with L1, Lcv and Lskew
Lmoms(AM.27090$Flow)
#store linear moments in an object
LPars <- as.numeric(Lmoms(AM.27090$Flow))[c(1,5,6)]
GEVPars(L1 = LPars[1], LCV = LPars[2], LSKEW = LPars[3])
```

GoTF — *Goodness of tail fit (GoTF).*

Description

Provides two GoTF scores for the generalised extreme value (GEV), Gumbel, generalised Pareto (GenPareto), or generalised logistic (GenLog) distribution. Also for any simulated numeric distribution

Usage

```
GoTF(x, dist = "GenLog", pars = NULL, GF = NULL, RepDist = NULL)
```

Arguments

`x`	a numeric vector. The sample of interest
`dist`	a choice of distribution to be assessed. The choices are "GenLog", "GEV", "Gumbel", or "GenPareto". The default is "GenLog"
`pars`	numeric vector of parameters for the GEV, GenLog, Gumbel, or GenPareto distributions. In the order of location, scale, shape (excluding shape for Gumbel)
`GF`	numeric vector of length three which are the growth factor statistics & QMED, in the order of Lcv, Lskew, & QMED
`RepDist`	a simulated sample (ideally of size = 5000*n) of a representative distribution to compare to the sample of interest

Details

The closer the results are to one, the better the fit. The GoTF is calculated by simulating the sample 5000 times with the desired distribution and calculating a statistic (in this case the coefficient of variation (CV) & mean) for the upper 25 percent of each sample. The same is calculated for the subject sample and compared to the distribution. The number of statistics from the simulated samples that are greater than the sample statistics is divided by 5000. The GoTF is this latter number where it is <0.5 and 1 minus this latter number where is it >0.5. If any further distributions are of interest, the representative distribution (RepDist) argument can be used. In this case a simulation of

5000*length(x) from that distribution can be used as RepDist, in place of using the dist input. If a sample that is not equal to 5000 time length(x) is in the RepDist argument, it will be resampled with replacement. An alternative is to use the pars or GF arguments which simulate from the distribution choice (dist) based on the parameters (location, scale, shape) or the growth factor (GF) inputs; the median annual maximum flow (QMED), linear coefficient of variation (Lcv), and linear skewnes (LSkew). The resulting probabilities for each statistic (the GoTF score) represent the probability of observing that statistic if the sample distribution has the same underlying distribution as the one under scrutiny.

Value

A data.frame with one row of probabilities representing the GoTF. The first column is the Tail cv and the second is the tail mean.

Author(s)

Anthony Hammond

Examples

```
#Get an AMAX sample and derive GoTF score against the GenLog and the GEV distributions
AM <- GetAM(203018)
GoTF(AM$Flow, dist = "GenLog")
GoTF(AM$Flow, dist = "GEV")
#Derive the GF parameters for the ungauged pooled estimate for the AM and
#calculate a GoTF for GenLog (assuming the gauged QMED)
#For this assume 0.16 and 0.2 as the ungauged Lcv & LSkew pooled estimates
 GoTF(AM$Flow, GF = c(0.16, 0.2, median(AM$Flow)))
#calculate the GoTF based on parameters of the GenLog estimated inadequately.
Loc <- mean(AM$Flow)
Scale <- sd(AM$Flow)
Skew <- 1-(median(AM$Flow)/mean(AM$Flow))
GoTF(AM$Flow, pars = c(Loc, Scale, Skew))
```

GoTFPool *Goodness of tail fit (GoTF) for pooling groups*

Description

Calculates GoTF scores for pooling groups for both generalised extreme value (GEV), generalised logistic (GenLog) & Gumbel distributions

Usage

```
GoTFPool(x)
```

Arguments

x	pooling group derived from the Pool function

Details

The GoTF for pooling groups is calculated by standardising all the sites in the group (dividing by median) and calculating the linear coefficient of variation (Lcv) and linear skewness (Lskew) of the pooled data as if it was one sample. The GoTF() function is then applied to the pooled data with the GF arguments using the aforementioned Lcv and Lskew, and QMED equal to one. The GoTF scores are calculated for the GEV, Gumbel, and GenLog distributions and can be used to assist the decision of which distribution to use for the final estimates. See details for the GoTF function for information about the resulting values. The closer the scores are to one, the better the tail fit.

Value

A list of two data.frames. Each with one row of the two GoTF values related to the columns; p(Tail cv) & p(Tail mean). See GoTF details. The first data.frame is for the GEV distribution, the second is for the GenLog distribution, and the third is for the Gumbel distribution.

Author(s)

Anthony Hammond

Examples

```
#Get CDs, create pooled group and calculate GoTFs.
CDs.203018 <- GetCDs(203018)
Pool.203018 <- Pool(CDs.203018)
GoTFPool(Pool.203018)
```

GumbelAM *Gumbel distribution - estimates directly from sample*

Description

Estimated quantiles as a function of return period (RP) and vice versa, directly from the data

Usage

```
GumbelAM(x, RP = 100, q = NULL, trend = FALSE)
```

Arguments

x	numeric vector (block maxima sample)
RP	return period (default = 100)
q	quantile (magnitude of variable)
trend	logical argument with default of FALSE. If TRUE, a linear adjustment to the location parameter is made to account for non-stationarity

Details

If the argument q is used, it overrides RP and provides RP as a function of q (magnitude of variable) as opposed to q as a function of RP. The parameters are estimated by the method of L-moments, as detailed in 'Hosking J. Wallis J. 1997 Regional Frequency Analysis: An Approach Based on L-moments. Cambridge University Press, New York'. The trend argument allows the location parameter to move in line with the observed linear trend of the sample. Another option is to detrend the sample first with the DeTrend function. On average this makes little difference to the two year flow but lower results for longer return periods (not always) when compared to the trend option in this function.

Value

quantile as a function of RP or vice versa, with the option of accounting for the linear trend in the sample

Author(s)

Anthony Hammond

Examples

```
#Get an annual maximum sample and estimate the 50-year RP
AM.27090 <- GetAM(27090)
GumbelAM(AM.27090$Flow, RP = 50)
#Estimate the RP for a 600m3/s discharge
GumbelAM(AM.27090$Flow, q = 600)
#Estimate the 50-year RP allowing for non-stationarity in the location parameter
GumbelAM(AM.27090$Flow, RP = 50, trend = TRUE)
```

`GumbelEst` *Gumbel distribution estimates from parameters*

Description

Estimated quantiles as function of return period (RP) and vice versa, from user input parameters

Usage

```
GumbelEst(loc, scale, q = NULL, RP = 100)
```

Arguments

loc	location parameter
scale	scale parameter
q	quantile. magnitude of the variable under consideration
RP	return period

Details

If the argument q is used, it overrides RP and provides RP as a function of q (magnitude of variable) as opposed to q as a function of RP.

Value

quantile as a function of RP or vice versa

Author(s)

Anthony Hammond

Examples

```
#Get an annual maximum sample, estimate the parameters and estimate 50-year RP
AM.27090 <- GetAM(27090)
Pars <- as.numeric(GumbelPars(AM.27090$Flow))
GumbelEst(Pars[1], Pars[2], RP = 50)
#Estimate the RP for a 600m3/s discharge
GumbelEst(Pars[1], Pars[2], q = 600)
```

GumbelGF *Gumbel distribution growth factors*

Description

Estimated growth factors as a function of return period, with inputs of Lcv & LSkew (linear coefficient of variation & linear skewness)

Usage

```
GumbelGF(lcv, RP)
```

Arguments

lcv	linear coefficient of variation
RP	return period

Details

Growth factors are calculated by the method outlined in the Flood Estimation Handbook, volume 3, 1999.

Value

Gumbel estimated growth factor

Author(s)

Anthony Hammond

Examples

```
#Estimate the 50-year growth factors from an Lcv of 0.17.
GumbelGF(0.17, RP = 50)
```

GumbelPars *Gumbel distribution parameter estimates*

Description

Estimated parameters from a sample (with Lmoments or maximum likelihood estimation) or from L1 (first L-moment), Lcv (linear coefficient of variation)

Usage

```
GumbelPars(x = NULL, mle = FALSE, L1, LCV)
```

Arguments

x	numeric vector. The sample
mle	logical argument with a default of FALSE. If FALSE the parameters are estimated with Lmoments, if TRUE the parameters are estimated by maximum likelihood estimation
L1	first Lmoment
LCV	linear coefficient of variation

Details

The L-moment estimated parameters are by the method detailed in 'Hosking J. Wallis J. 1997 Regional Frequency Analysis: An Approach Based on L-moments. Cambridge University Press, New York'

Value

Parameter estimates (location, scale)

Author(s)

Anthony Hammond

Examples

```
#Get an annual maximum sample and estimate the parameters using Lmoments
AM.27090 <- GetAM(27090)
GumbelPars(AM.27090$Flow)
#Estimate parameters using MLE
GumbelPars(AM.27090$Flow, mle = TRUE)
#calculate Lmoments and estimate the parmeters with L1 and Lcv
Pars <- as.numeric(Lmoms(AM.27090$Flow)[c(1,5)])
GumbelPars(L1 = Pars[1], LCV = Pars[2])
```

H2 — *Heterogeneity measure (H2) for pooling groups.*

Description

Quantifies the heterogeneity of a pooled group

Usage

```
H2(x)
```

Arguments

x	pooling group derived from the Pool() function

Details

The H2 measure was developed by Hosking & Wallis and can be found in their book 'Regional Frequency Analysis: an approach based on LMoments (1997). It was also adopted for use by the Flood Estimation Handbook (1999) and is described in volume 3.

Value

A vector of two characters; the first representing the H2 score and the second stating a qualitative measure of heterogeneity.

Author(s)

Anthony Hammond

Examples

```
#Get CDs, form a pooling group and calculate H2
CDs.203018 <- GetCDs(203018)
Pool.203018 <- Pool(CDs.203018)
H2(Pool.203018)
```

HydroPlot *Hydrological plot of concurrent discharge and precipitation*

Description

Plots concurrent precipitation and discharge with precipitation along the top and discharge along the bottom

Usage

```
HydroPlot(
  x,
  Title = "Concurrent Rainfall & Discharge",
  from = NULL,
  to = NULL,
  adj.y = 1.5,
  plw = 1,
  qlw = 1.8
)
```

Arguments

`x`	a data.frame with three columns in the order of date (or POSIXct), precipitation, and discharge
`Title`	a character string. The user chosen plot title. The default is "Concurrent Rainfall & Discharge"
`from`	a starting time for the plot. In the form of a date or POSIXct object. The default is the first row of x
`to`	an end time for the plot. In the form of a date or POSIXct object. The default is the last row of x
`adj.y`	a numeric value to adjust the closeness of the preciptation and discharge in the plot. Default is 1.5. A lower value brings them closer and a larger value further apart
`plw`	a numeric value to adjust the width of the precipitation lines. Default is one. A larger value thickens them and vice versa
`qlw`	a numeric value to adjust the width of the discharge line. Default is 1.8. A larger value thickens them and vice versa

Details

The input of x is a dataframe with the first column being time. If the data is sub daily this should be class POSIXct with time as well as date.

Value

A plot of concurrent precipitation and discharge. With the former at the top and the latter at the bottom.

Author(s)

Anthony Hammond

Examples

```
#Plot the Thames precipitation and discharge for the 2013 hydrological year,
#adjusting the y axis to 1.8.
HydroPlot(ThamesPQ, from = "2013-10-01", to = "2014-09-30", adj.y = 1.8)
```

`ImportAM` *Import an annual maximum (AMAX) sample from NRFA peak flow .AM files*

Description

Imports the peak flows and dates from from NRFA peak flow .AM files, exluding the rejected years

Usage

```
ImportAM(x)
```

Arguments

x	the file path for the .AM file

Details

File paths for importing data require forward slashes. On some operating systems, such as windows, the copy and pasted file paths will have backward slashes and would need to be changed accordingly.

Value

A data.frame with columns; Date and Flow

Author(s)

Anthony Hammond

Examples

```
#Import an AMAX sample and display the first six rows in the console
## Not run: AM.4003 <- ImportAM("C:/Data/NRFAPeakFlow_v10/Suitable for QMED/4003.AM")
## Not run: head(AM.4003)]
```

ImportCDs *Import catchment descriptors from .CD3 files*

Description

Imports catchment descriptors from CD3 files either from an FEH webservice download or from the Peakflows dataset downloaded from the national river flow archive (NRFA) website

Usage

```
ImportCDs(x)
```

Arguments

x	the CD3 file path

Details

The CD3 files downloaded from the FEH webserver are formatted differently from the CD3 files of the peak flows dataset, this function is coded to import either, given the correct file path. File paths for importing data require forward slashes. On some operating systems, such as windows, the copy and pasted file paths will have backward slashes and would need to be changed accordingly.

Value

A data.frame with columns; Descriptor and Value.

Author(s)

Anthony Hammond

Examples

```
#Import catchment descriptors from a NRFA peakflows CD3 file and display in console
## Not run: CDs.4003 <- ImportCDs("C:/Data/NRFAPeakFlow_v9/Suitable for QMED/4003.CD3", web = FALSE)
## Not run: CDs.4003
#Import catchment descriptors from a FEH webserver CD3 file and display CDs in the console
## Not run: CDs.MySite <- ImportCDs("C:/Data/FEH_Catchment_384200_458200.CD3")
```

Lcv — *Linear coefficient of variation (Lcv)*

Description

Calculates the Lcv from a sample of data

Usage

```
Lcv(x)
```

Arguments

x	a numeric vector. The sample of interest

Details

Lcv calculated according to methods outlined by Hosking & Wallis (1997): Regional Frequency Analysis and approach based on LMoments. Also in the Flood Estimation Handbook (1999), volume 3.

Value

Numeric. The Lcv of a sample.

Author(s)

Anthony Hammond

Examples

```
#Get an AMAX sample and calculate the Lmoments
AM.96001 <- GetAM(96001)
Lcv(AM.96001$Flow)
```

LcvUrb — *Urban adjustment for the linear coefficient of variation (Lcv)*

Description

Urbanises or de-urbanises the Lcv using the methods outlined in the guidance by Wallingford HydroSolutions: 'WINFAP 4 Urban Adjustment Procedures'

Usage

```
LcvUrb(lcv, URBEXT2000, DeUrb = FALSE)
```

Arguments

`lcv`	the Lcv (numeric)
`URBEXT2000`	quantiication of urban and suburbanisation for the subject catchment
`DeUrb`	logical argument with a default of FALSE. If set to TRUE, de-urbanisation adjustment is performed, if FALSE, urbanisation adjustment is performed

Details

The method for de-urbanisation isn't explicitly provided in 'WINFAP 4 Urban Adjustment Procedures', but the procedure is a re-arrangment of the urbanisation equation, solving for Lcv rather than Lcv-urban.

Value

The urban adjust Lcv or the de-urbanised Lcv

Author(s)

Anthony Hammond

Examples

```
#Choose an urban site (site 53006) from the NRFA data then apply a de-urban
#adjustment using the Lcv and URBEXT2000 displayed
NRFAData[which(rownames(NRFAData) == 53006),]
LcvUrb(0.21, 0.1138, DeUrb = TRUE)
#Get the pooled Lmoment ratios results for catchment 53006 and apply the
#urban adjustment using the pooled Lcv, and the URBEXT2000 for site 53006.
CDs.53006 <- GetCDs(53006)
QuickResults(CDs.53006)[[2]]
LcvUrb(0.196, 0.1138)
```

`LKurt` *Linear Kurtosis (LKurt)*

Description

Calculates the LKurtosis from a sample of data

Usage

```
LKurt(x)
```

Arguments

`x`	a numeric vector. The sample of interest

Details

LKurtosis calculated according to methods outlined by Hosking & Wallis (1997): Regional Frequency Analysis and approach based on LMoments. Also in the Flood Estimation Handbook (1999), volume 3.

Value

Numeric. The LSkew of a sample.

Author(s)

Anthony Hammond

Examples

```
#Get an AMAX sample and calculate the Lmoments
AM.96001 <- GetAM(96001)
LKurt(AM.96001$Flow)
```

Lmoms *Lmoments & Lmoment ratios*

Description

Calculates the Lmoments and Lmoment ratios from a sample of data

Usage

```
Lmoms(x)
```

Arguments

x	a numeric vector. The sample of interest

Details

Lmoments calculated according to methods outlined by Hosking & Wallis (1997): Regional Frequency Analysis and approach based on LMoments. Also in the Flood Estimation Handbook (1999), volume 3.

Value

A data.frame with one row and column headings; L1, L2, L3, L4, Lcv, LSkew, and LKurt. The first four are the Lmoments and the next three are the Lmoment ratios.

Author(s)

Anthony Hammond

Examples

```
#Get an AMAX sample and calculate the Lmoments
AM.96001 <- GetAM(96001)
Lmoms(AM.96001$Flow)
```

`LRatioChange` *Adjust L-Ratios in a pooling group*

Description

Adjusts the linear coefficient of variation (Lcv) and the linear skewness (LSkew) for a chosen site in a pooling group

Usage

```
LRatioChange(x, SiteID, lcv, lskew)
```

Arguments

`x`	pooling group derived with the Pool function
`SiteID`	the identification number of the site in the pooling group that is to be changed (character or integer)
`lcv`	The user supplied Lcv. numeric
`lskew`	The user supplied LSkew. numeric

Details

Pooling groups are formed from the NRFAData data.frame and all the Lcv and LSkew values are precalculated using the National River Flow Archive Peak flow dataset noted in the description file. The resulting pooled growth curve is calculated using the Lcv and Lskew in the pooled group. The user may have further data and be able to add further peak flows to the annual maximum samples within a pooling group. If that is the case a new Lcv and Lskew can be determined using the Lmoms function. These new values can be added to the pooling group with this LRatioChange function. Also the permeable adjustment function may have been applied to a site, which provides a new Lcv and LSkew. In which case, the LRatioChange function can be applied. The function creates a new pooling group object and x will still exist in it's original state after the function is applied.

Value

A new pooling group, the same as x except for the user adjusted Lcv and Lskew for the user selected site.

Author(s)

Anthony Hammond

Examples

```
# Get some catchment descriptors and create a pooling group.
CDs.39001 <- GetCDs(39001)
Pool.39001 <- Pool(CDs.39001, iug = TRUE)
# apply the function to create a new adjusted pooling group,
#changing the subject site lcv and lskew to 0.187 and 0.164, respectively
Pool.39001Adj <- LRatioChange(Pool.39001, SiteID = 39001, lcv = 0.187, lskew = 0.164)
```

LSkew *Linear Skewness (LSkew)*

Description

Calculates the LSkew from a sample of data

Usage

```
LSkew(x)
```

Arguments

x	a numeric vector. The sample of interest

Details

LSkew calculated according to methods outlined by Hosking & Wallis (1997): Regional Frequency Analysis and approach based on LMoments. Also in the Flood Estimation Handbook (1999), volume 3.

Value

Numeric. The LSkew of a sample.

Author(s)

Anthony Hammond

Examples

```
#Get an AMAX sample and calculate the Lmoments
AM.96001 <- GetAM(96001)
LSkew(AM.96001$Flow)
```

LSkewUrb	*Urban adjustment for the linear skewness (LSkew)*

Description

Urbanises or de-urbanises the LSkew using the methods outlined in the guidance by Wallingford HydroSolutions: 'WINFAP 4 Urban Adjustment Procedures'

Usage

```
LSkewUrb(lskew, URBEXT2000, DeUrb = FALSE)
```

Arguments

`lskew`	the LSkew (numeric)
`URBEXT2000`	quantiication of urban and suburbanisation for the subject site
`DeUrb`	logical argument with a default of FALSE. If set to TRUE, de-urbanisation adjustment is performed, if FALSE, urbanisation adjustment is performed

Details

The method for de-urbanisation isn't explicitly provided in 'WINFAP 4 Urban Adjustment Procedures', but the procedure is a re-arrangment of the urbanisation equation, solving for LSkew rather than LSkew-urban.

Value

The urban adjust Lcv or the de-urbanised Lcv

Author(s)

Anthony Hammond

Examples

```
#Choose an urban site (site 53006) from the NRFA data then apply a de-urban
#adjustment using the Lcv and URBEXT2000 displayed
NRFAData[which(rownames(NRFAData) == 53006),]
LSkewUrb(0.124, 0.1138, DeUrb = TRUE)
#Get the pooled Lmoment ratios results for catchment 53006 and apply the urban
#Get the CDS & adjustment using the pooled LSkew, and the URBEXT2000 for site 53006.
CDs.53006 <- GetCDs(53006)
QuickResults(CDs.53006)[[2]]
LSkewUrb(0.194, 0.1138)
```

NGRDist — *British national grid reference (NGR) distances*

Description

Calculates the euclidean distance between two british national grid reference points using the pythagorean method

Usage

```
NGRDist(i, j)
```

Arguments

`i`	a numeric vector of length two. The first being the easting and the second being the northing of the first site
`j`	a numeric vector of length two. The first being the easting and the second being the northing of the second site

Details

Note, that the result is converted to km when six digits are used for easting and northing, when six digits would usually provide a result in metres.

Value

A distance in kilometres (if six digits for easting and northing are used)

Author(s)

Anthony Hammond

Examples

```
#Calculate the distance between the catchment centroid for the
#Kingston upon Thames river gauge and the catchment centroid for the
#gauge at Ardlethen on the River Ythan. First view the eastings and northings
GetCDs(10001)
GetCDs(39001)
NGRDist(i = c(381355, 839183), j = c(462899, 187850))
```

`NRFAData` *National River Flow Archive descriptors and calculated statistics for sites suitable for pooling*

Description

A data.frame of catchment descriptors, Lmoments, Lmoment ratios, sample size and median annual maximum flow (QMED). NRFA Peak Flow Dataset - Version 10.

Usage

```
NRFAData
```

Format

A data frame with 546 rows and 27 variables

Details

The functions for pooling group formation and estimation rely on this dataframe. However, the data frame is open for manipulation in case the user wishes to add sites that aren't included, or change parts where local knowledge has improved on the data. Although, usually, in the latter case, such changes will be more appropriately applied to the formed pooling group. If changes are made, they will only remain within the workspace. If a new workspace is opened and the UKFE package is loaded, the data frame will have returned to it's original state.

Source

https://nrfa.ceh.ac.uk/peak-flow-dataset

`OptimPars` *Optimise distribution parameters*

Description

Estimates the parameters of the generalised extreme value, generalised logistic, or Gumbel distribution from known return period estimates

Usage

```
OptimPars(x, dist = "GenLog")
```

Arguments

`x`	a data.frame with RPs in the first column and associated estimates in the second column
`dist`	a choice of distribution for the estimates. The choices are "GenLog", "GEV", or "Gumbel" - the generalised logistic, generalised extreme value and Gumbel distribution, respectively. The default is "GenLog"

Details

Given a dataframe with return periods (RPs) in the first column and associated estimates in the second column, this function provides an estimate of the distribution parameters. Ideally the first RP should be 2. Extrapolation outside the RPs used for calibration comes with greater uncertainty.

Value

The location, scale and shape parameters for the generalised logistic or Generalised extreme value distribution. Or the location and scale for the Gumbel.

Author(s)

Anthony Hammond

Examples

```
#Get some catchment descriptors and some quick results. Then estmate the GenLog parameters
Results <- QuickResults(GetCDs(96001), plot = FALSE)[[1]]
OptimPars(Results[,1:2])
```

`PermAdj` *Permeable Adjustment*

Description

Adjusts the linear coefficient of variation (Lcv) and the linear skewness to account for non-flood years

Usage

```
PermAdj(x)
```

Arguments

`x`	The annual maximum sample. Numeric vector

Details

The permeable adjustment method is detailed in chapter 19, volume three, of the Flood Estimation Handbook, 1999. The method makes no difference for sites where there are no annual maximums (AM) in the sample that are < median(AM)/2. Once applied the results can be used with the LRatioChange function to update the associated member of a pooling group. Or can be applied directly with the growth factor functions for a single site estimate.

Value

A dataframe with one row and two columns. Lcv in the first column and Lskew in the second

Author(s)

Anthony Hammond

Examples

```
# Get an anuual maximum sample with a BFIHOST above 0.65 and with some
# annual maximums lower than median(AM)/2. And then apply the function.
AM.39001 <- GetAM(39001)
PermAdj(AM.39001$Flow)
```

Pool *Create pooling group*

Description

Function to develop a pooling group based on catchment descriptors

Usage

```
Pool(
  CDs = NULL,
  AREA,
  SAAR,
  FARL,
  FPEXT,
  N = 500,
  exclude = NULL,
  iug = FALSE,
  DeUrb = FALSE
)
```

Arguments

`CDs`	catchment descriptors derived from either GetCDs or ImportCDs
`AREA`	catchment area in km2
`SAAR`	catchment standard average annual rainfall (1961-1990) in mm
`FARL`	catchment flood attenuation from reservoirs & lakes
`FPEXT`	catchment floodplain extent. The proportion of the catchment that is estimated to be inundated by a 100-year flood
`N`	minimum Number of total gauged record years for the pooling group
`exclude`	sites to exclude from the pooling group. Either a single site reference or a vector of site references (numeric)
`iug`	iug stands for 'include urban gauge'. It's a logical argument with default of FALSE. TRUE will over-ride the default and add the closest site in catchment descriptor space to the CDs provided if it has URBEXT2000 >= 0.03
`DeUrb`	logical argument with a default of FALSE. If true, the Lcv and LSkew of the top site in the pooling group will be de-urbanised

Details

A pooling group is created from a CDs object, derived from GetCDs or ImportCDs, or specifically with the catchment descriptors AREA, SAAR, FARL, & FPEXT (see arguments). To change the default pooling group one or more sites can be excluded using the 'exclude' option, which requires either a site reference or multiple site references in a vector. If this is done, the site with the next lowest similarity distance measure is added to the group (until the total number of years is at least N). Sites with URBEXT2000 (urban extent) > 0.03 are excluded by default. If a gauged assessment is required and the site of interest is urban it can be included by setting iug = TRUE. In which case the user may wish to de-urbanise the subject site's Lcv and Lskew (L-moment ratios) by setting DeUrb = TRUE. If the user has more data available for a parcticular site within the pooling group, the Lcv and Lskew for the site can be updated after the group has been finalised. An example of doing so is provided below. The pooling method is outlined in Science Report: SC050050 - Improving the FEH statistical procedures for flood frequency estimation.

Value

A data.frame of the pooling group with site refence row names and 24 columns each providing catchment & gauge details for the sites in the pooling group.

Author(s)

Anthony Hammond

Examples

```
#Get some catchment descriptors
CDs.73005 <- GetCDs(73005)
#Set up a pooling group object called Pool.73005 excluding sites 79005 & 71011.
#Then print the group to the console
Pool.73005 <- Pool(CDs.73005, exclude = c(79005, 71011))
```

```
Pool.73005
#Form a pooling group, called PoolGroup, with the catchment descriptors specifically
PoolGroup <- Pool(AREA = 1000, SAAR = 800, FARL = 1, FPEXT = 0.01)
#Form a pooling group using an urban catchment which is intended for enhanced
#single site estimation - by including it in the group.
CDs.39001 <- GetCDs(39001)
Pool.39001 <- Pool(CDs.39001, iug = TRUE, DeUrb = TRUE)
#Change the Lcv and LSkew of the top site in the pooling group to 0.19 & 0.18,
#respectively. Lcv and Lskew are in columns 16 & 17.
Pool.39001[1,c(16, 17)] <- c(0.19, 0.18)
```

PoolEst *Pooled flood estimates*

Description

Provides pooled results from a pooling group - gauged, ungauged and with urban adjustment if necessary.

Usage

```
PoolEst(
  x,
  gauged = FALSE,
  QMED,
  dist = "GenLog",
  RP = c(2, 5, 10, 20, 50, 75, 100, 200, 500, 1000),
  UrbAdj = FALSE,
  CDs = NULL,
  URBEXT = NULL,
  trend = FALSE,
  fseQMED = 1.41
)
```

Arguments

x	pooling group derived from the Pool function
gauged	logical argument with a default of FALSE. TRUE for gauged results and FALSE for ungauged
QMED	estimate of the median annual maximum flow
dist	a choice of distribution for the estimates. The choices are "GenLog", "GEV", or "Gumbel"; the generalised logistic, generalised extreme value, and Gumbel distribution, respectively. The default is "GenLog"
RP	return period of interest. By default the following RPs are provided: 2, 5, 10, 20, 50, 75, 100, 200, 500, 1000
UrbAdj	logical argument with a default of FALSE. When TRUE, an urban adjustment is applied to the pooled Lcv and LSkew

`CDs`	catchment descriptors derived from either GetCDs or ImportCDs
`URBEXT`	the catchment URBEXT2000, to be supplied if UrbAdj is TRUE and if CDs have not been
`trend`	logical argument with a default of FALSE. TRUE adjusts the stationary QMED estimate to a non-stationary estimate
`fseQMED`	factorial standard error of the median annual maximum (QMED) estimate, used for quantifying ungauged uncertainty. Default is 1.41

Details

PoolEst is a function to provide results from a pooling group derived using the Pool function. QMED (median annual maximum flow) needs to be supplied and can be derived from the QMED function for ungauged estimates or the annual maximum sample for gauged estimates. If the catchment of interest is urban, the UrbAdj argument can be set to TRUE. If this is done, either URBEXT (urban extent) needs to be provided or the catchment descriptors, derived from ImportCDs or GetCDs. The methods for estimating pooled growth curves are according to Science Report: SC050050 - Improving the FEH statistical procedures for flood frequency estimation. The methods for estimating the L-moments and growth factors are outlined in the Flood Estimation Handbook (1999), volume 3. The methods for quantifying uncertainty are detailed in Hammond, A. (2021). Sampling uncertainty of UK design flood estimation. Hydrology Research, 52 (6): 1357–1371. When UrbAdj = TRUE, urban adjustment is applied to the QMED estimate according to the method outlined in the guidance by Wallingford HydroSolutions: 'WINFAP 4 Urban Adjustment Procedures'. If trend = TRUE & gauged = FALSE, the QMED is multiplied by a trend coefficient. The coefficient was derived by calculating a weighted (by sample size) mean proportional change in the 2-year flow, from the UK National River Flow Archive benchmark sites considered suitable for pooling. The weighted per year change was first calculated and then multiplied by half the mean sample size of sites suitable for QMED. If trend = TRUE and gauged = TRUE, the weighted per year change is multiplied by half the sample size of the first site in the pooling group. This approach attempts to include a generic non-stationarity in the pooled estimates by adjusting the location parameter. Among other assumptions (such as the change being linear and only to the location of the distribution), it makes the assumption that apparent trends at individual sites are unreliable due to short record lengths or site specific due to human influence, but across many sites the mean increase in median peak discharge is representative of a non-stationary process across the UK (this approach will be regionalised in a later version of the tool). The trend is applied to whatever is in the QMED argument. Therefore, trend should equal FALSE if the QMED estimate has been user adjusted for trend already. If Lcv & Lskew have also been user adjusted for a gauged site, these can be changed in the pooling group (see Pool function details). The per year, weighted mean QMED trend, was estimated to be 0.12 percent +/- 0.05 (95 percent uncertainty - calculated by weighted resampling).

Value

A list of length two. Element one is a data frame with columns; return period (RP), peak flow estimates (Q), growth factor estimates (GF), lower and upper intervals of uncertainty (68 percent intervals for ungauged and 95 percent for gauged). The second element is the estimated Lcv and Lskew.

Author(s)

Anthony Hammond

Examples

```
#Get some catchment descriptors and form a pooling group. It's urban and
#therefore the site of interest is not included.
CDs.27083 <- GetCDs(27083)
Pool.27083 <- Pool(CDs.27083)
#Get results for the ungauged case, with urban adjustment
PoolEst(Pool.27083, QMED = 11.941, UrbAdj = TRUE, CDs = CDs.27083)
#Form the group again with the urban gauge included & undertake a gauged estimate
#with urban adjustment. QMED in this example is estimated as the median of the annual
#maximum series for site 27083.
PoolG.27083 <- PoolG.27083 <- Pool(CDs.27083, iug = TRUE, DeUrb = TRUE)
PoolEst(PoolG.27083, QMED = 12.5, UrbAdj = TRUE, CDs = CDs.27083)
```

POTextract — *Peaks over threshold (POT) data extraction*

Description

Extracts independent peaks over a threshold from a sample

Usage

```
POTextract(x, div = NULL, thresh = 0.975, Plot = TRUE)
```

Arguments

`x`	either a numeric vector or dataframe with date (or POSIXct) in the first column and hydrological variable in the second
`div`	user chosen value, either side of which two peaks over the threshold are considered independent. Default is the mean of the sample
`thresh`	user chosen threshold. Default is 0.975
`Plot`	logical argument with a default of TRUE. When TRUE, the full hydrograph with the peaks over the threshold highlighted is plotted

Details

If the x argument is a numeric vector, the peaks will be extracted with no time information. x can instead be a data.frame with dates in the first column and the numeric vector in the second. In this latter case, the peaks will be timestamped and a hydrograph including POT will be plotted by default. The method of extracting independent peaks assumes that there is a value either side of which, events can be considered independent. For example, if two peaks above the chosen threshold are separated by the daily mean flow, they could be considered independent, but not if flow hasn't returned to daily mean at any time between the peaks. Daily mean flow may not always be appropriate, in which case the 'div' argument can be adjusted. The function was coded primarily for river flow but for extracting daily duration POT rainfall a div of zero could be used (making the assumption that rainfall events separated by a period of 24 hours, with no rain, are independent). For sub-daily rainfall, further work, after use of the function, would be necessary. For example, a div of zero could be used, and if two peaks are extracted but not separated by more than 24 hours, the lower of the two could be discarded. For this approach a data.frame with dates would be required. When plotted, the blue line is the threshold and the green line is the independence line (div).

Value

Prints the number of peaks per year and returns a data.frame with columns; Date and peak, with the option of a plot. Or a numeric vector of peaks is returned if only a numeric vector of the hydrological variable is input.

Author(s)

Anthony Hammond

Examples

```
#Extract POT data from Thames mean daily flow 1970-10-01 to 2015-09-25 with
#div = mean and threshold = 0.95. Then display the first six rows
ThamesQPOT <- POTextract(ThamesPQ[, c(1,3)], thresh = 0.90)
head(ThamesQPOT)
#Extract Thames POT from only the numeric vector of flows and display the
#first six rows
ThamesQPOT <- POTextract(ThamesPQ[, 3], thresh = 0.90)
ThamesQPOT
#Extract the Thames POT precipitation with a div of 0 and the default
#threshold. Then display the first six rows
ThamesPPOT <- POTextract(ThamesPQ[, c(1,2)], div = 0)
head(ThamesPPOT)
```

`QMED` *QMED (median annual maximum flow) estimate from catchment descriptors*

Description

Estimated median annual maximum flow from catchment descriptors and donor sites

Usage

```
QMED(
  CDs = NULL,
  Don1 = NULL,
  Don2 = NULL,
  UrbAdj = FALSE,
```

```
  AREA,
  SAAR,
  FARL,
  BFIHOST,
  URBEXT2000 = NULL
)
```

Arguments

CDs	catchment descriptors derived from either GetCDs or ImportCDs
Don1	numeric site reference for the a signle donor (for donor candidates see DonAdj function)
Don2	vector of two site references for two donors (for donor candidates see DonAdj function)
UrbAdj	logical argument with a default of FALSE. True applies an urban adjustment
AREA	catchment area in km2
SAAR	standard average annual rainfall (mm)
FARL	flood attenuation from reservoirs and lakes
BFIHOST	baseflow index calculated from the catchment hydrology of soil type classification
URBEXT2000	measure of catchment urbanisation

Details

QMED is estimated from catchment descriptors: QMED = 8.3062*AREA^0.8510 0.1536^(1000/SAAR) FARL^3.4451 0.0460^(BFIHOST^2) as derived in Science Report: SC050050 - Improving the FEH statistical procedures for flood frequency estimation. The single donor method is from the same paper. The method for two donors is outlined in 'Kjeldsen, T. (2019). Adjustment of QMED in ungauged catchments using two donor sites. Circulation - The Newsletter of the British Hydrological Society, 4'. When UrbAdj = TRUE, urban adjustment is applied to the QMED estimate according to the method outlined in the guidance by Wallingford HydroSolutions: 'WINFAP 4 Urban Adjustment Procedures'. For flexibility there is the option to input the relevant catchment descriptors directly rather than a CDs object.

Value

An estimate of QMED from catchment descriptors

Author(s)

Anthony Hammond

Examples

```
#Get some catchment descriptors and calculate QMED as if it was ungauged, with
#no donors, one donor, and two donors
CDs.55004 <- GetCDs(55004)
```

```
QMED(CDs.55004)
QMED(CDs.55004, Don1 = 55012)
QMED(CDs.55004, Don2 = c(55012, 60007))
#Get CDs for urban gauge and calculate QMED with urban adjustment
CDs.27083 <- GetCDs(27083)
QMED(CDs.27083, UrbAdj = TRUE)
```

QMEDData *National River Flow Archive descriptors and calculated statistics for sites suitable for QMED & pooling*

Description

A data.frame of catchment & data descriptors relating to the median annual maximum flow (QMED). NRFA Peak Flow Dataset - Version 10

Usage

```
QMEDData
```

Format

A data frame with 885 rows and 24 variables

Details

The functions for QMED estimation and retreieval of catchment descriptors rely on this dataframe. However, the data frame is open for manipulation in case the user wishes to add sites that aren't included, or change parts where local knowledge has improved on the data. If changes are made, they will only remain within the workspace. If a new workspace is opened and the UKFE package is loaded, the data frame will have returned to it's original state.

Source

https://nrfa.ceh.ac.uk/peak-flow-dataset

QMEDDonEq *QMED donor adjustment*

Description

Applies a donor adjustment to the median annual maximum flow (QMED) estimate

Usage

```
QMEDDonEq(
  AREA,
  SAAR,
  FARL,
  BFIHOST,
  QMEDgObs,
  QMEDgCds,
  xSI,
  ySI,
  xDon,
  yDon,
  alpha = TRUE
)
```

Arguments

AREA	catchment area in km2
SAAR	standardised average annual rainfall in mm
FARL	flood attenuation from reservoirs and lakes
BFIHOST	the baseflow index as a function of soil type
QMEDgObs	the observed QMED at the donor site
QMEDgCds	the QMED equation derived QMED at the donor site
xSI	the catchment centroid easting for the site of interest
ySI	the catchment centroid northing for the site of interest
xDon	the catchment centroid easting for the donor site
yDon	the catchment centroid northing for the donor site
alpha	a logical argument with a default of TRUE. When FALSE the exponent in the donor equation is set to one. Otherwise it is determined by the distance between the donor and the subject site

Details

Although a single donor adjustment can be applied with the DonAdj() function and the QMED(), this is provided for flexibility. The method is that of Science Report: SC050050 - Improving the FEH statistical procedures for flood frequency estimation (2008).

Author(s)

Anthony Hammond

Examples

```
#Get observed QMED for site 96003
Qob <- median(GetAM(96003)[,2])
#Get QMED equation estimated QMED for the donor site
```

```
QCD <- QMED(CDs = GetCDs(96003))
#display CDs for site 96001 & note the easting and northing
GetCDs(96001)
#display CDs for site 96003 & note the easting and northing
GetCDs(96003)
#Apply the QMEDDonEq function with the information gained
QMEDDonEq(194, 1096, 0.955, 0.297, Qob, QCD, xSI = 289289,ySI = 947523,xDon = 280908,yDon = 953653)
```

QMEDfseSS *QMED factorial standard error for gauged sites*

Description

Estimates the median annual maximum flow (QMED) factorial standard error (FSE) by bootstrapping the sample

Usage

```
QMEDfseSS(x)
```

Arguments

x	a numeric vector. The sample of interest

Details

The bootstrapping procedure resamples from the sample N*500 times with replacement. After splitting into 500 samples of size N, the median is calculated for each. Then the exponent of the standard deviation of the log transformed residuals is taken as the FSE. i.e. exp(sd(log(x)-mean(log(x)))), where x is the bootstrapped medians.

Value

The factorial standard error for the median of a sample.

Author(s)

Anthony Hammond

Examples

```
#Extract an AMAX sample and estimate the QMED factorial standard error
AM.203018 <- GetAM(203018)
QMEDfseSS(AM.203018$Flow)
```

QMEDLink — *QMED Linking equation*

Description

Estimates the median annual maximum flow (QMED) from non-flood flows

Usage

```
QMEDLink(Q5dmf, Q10dmf, DPSBAR, BFI)
```

Arguments

Q5dmf	numeric. The daily mean flow that is exceeded 5 percent of the time
Q10dmf	numeric. The daily mean flow that is exceeded 10 percent of the time
DPSBAR	a catchment descriptor. The average drainage path slope of the catchment
BFI	the baseflow index of the gauged flow

Details

The QMED Linking equation estimates QMED as a function of the flow that is exceeded five percent of the time, the flow that is exceeded 10 percent of the time, the baseflow index, and the catchment desciptor; drainage path slope (DPSBAR). All of these can be found for sites on the National River Flow Archive (NRFA) website. The method is provided in the guidance note 'WINFAP 4 QMED Linking equation' (2016) by Wallingford HydroSolutions.

Author(s)

Anthony Hammond

Examples

```
#Calculate the QMED for site 1001 (Wick at Tarroul)
QMEDLink(10.14, 7.352, 29.90, 0.39)
```

QMEDPOT — *Empirical estimate of QMED from peaks over threshold (POT) data*

Description

Estimates the median annual maximum flow (QMED) from peaks over threshold data

Usage

```
QMEDPOT(x, ppy)
```

Arguments

`x`	numerical vector. POT data
`ppy`	number of peaks per year in the POT data

Details

If there are multiple peaks per year, the peaks per year (ppy) argument is used to convert to the annual scale to derive QMED. If ppy is one, then the median of the POT sample is returned (the median of x).

Author(s)

Anthony Hammond

Examples

```
#Extract some POT data and estimate QMED
ThamesPOT <- POTextract(ThamesPQ[,c(1,3)], thresh = 0.90)
QMEDPOT(ThamesPOT$peak, ppy = 1.867263)
```

`QuickResults` *Quick pooled results*

Description

Provides pooled gauged, ungauged, or fake ungauged results, directly from the catchment descriptors

Usage

```
QuickResults(
  CDs,
  gauged = FALSE,
  dons = 2,
  Qmed = NULL,
  trend = FALSE,
  FUngauged = FALSE,
  plot = TRUE,
  dist = "GenLog"
)
```

Arguments

`CDs`	catchment descriptors derived from either GetCDs or ImportCDs
`gauged`	logical argument with a default of FALSE. TRUE for gauged results and FALSE for ungauged
`dons`	number of donors required with a choice of 0, 1, or 2

`Qmed`	user supplied QMED which overrides the default QMED estimate
`trend`	logical argument with a default of FALSE. TRUE adjusts the stationary QMED estimate to a non-stationary estimate
`FUngauged`	logical argument with a default of FALSE. TRUE provides an ungauged estimate whilst excluding the gauged site (the site with the most similar CDs)
`plot`	logical argument with a default of TRUE. TRUE provides an extreme value plot. FALSE prevents the plot
`dist`	a choice of distribution for the estimates. The choices are "GenLog", "GEV", or "Gumbel; the generalised logistic, generalised extreme value, and Gumbel distributions, respectively. The default is "GenLog"

Details

The quick results function provides results with a default pooling group. If gauged = FALSE the median annual maximum flood (QMED) is estimated from catchment descriptors using the QMED equation and then adjusted with two of the closest un-urban gauged sites (can be changed to 0 or 1 donors). If the site is urban, an urban adjustment is made to the QMED and to the pooled growth curve. If gauged = TRUE QMED is the median of the gauged annual maxima and the growth curve is formed with the gauged weighting procedure (often known as enhanced single site). If the gauged catchment is urban, it's included in the pooling group and deurbanised before an urban adjustment is made to the final growth curve. If FUngauged = TRUE, the top site in the pooling group is excluded and the estimate is performed henceforth in the manner of gauged = FALSE. If the CDs are from a gauged site that is not in the list of sites that are considered suitable for pooling, it won't be included in the pooling group. In which case, if gauged = TRUE, the result, will be erroneous. For more details of the trend argument see the details for the PoolEst function.

Value

A list of length two. Element one is a data frame with columns; return period (RP), peak flow estimates (Q) and growth factor estimates (GF). Two additional columns quantify the uncertainty. The second element is the estimated Lcv and Lskew (linear coefficient of variation and skewness). By default an extreme value plot is also returned

Author(s)

Anthony Hammond

Examples

```
#Get some catchment descriptors
CDs.73005 <- GetCDs(73005)
#Get default ungauged results
QuickResults(CDs.73005)
#Get gauged results with a GEV distribution
QuickResults(CDs.73005, gauged = TRUE, dist = "GEV")
#Get fake ungauged results with one donor
QuickResults(CDs.73005, FUngauged = TRUE, dons = 1)
```

Rating *Stage-Discharge equation optimisation*

Description

Optimises a power law rating equation from observed discharge and stage

Usage

```
Rating(x, a = NULL)
```

Arguments

x	a data.frame with discharge in the first column and stage in the second
a	a user defined stage correction

Details

The power law rating equation optimised here has the form q = c(h+a)^n; where 'q' is flow, 'h' is the stage, c' and 'n' are constants, and 'a' is the stage when flow is zero. The optimisation uses all the data provided in the dataframe (x). If separate rating limbs are necessary, x can be subset per limb. i.e. the rating function would be used multiple times, once for each subset of x. There is the option, with the 'a' argument, to hold the stage correction parameter (a), at a user defined level. If 'a' is NULL it will be calibrated with 'c' & 'n' as part of the optimisation procedure.

Value

A list with three elements. The first is a vetor of the three calibrated rating parameters. The second is the rating equation; discharge as a function of stage. The third is the rating equation; stage as a function of discharge. A rating plot is also returned.

Author(s)

Anthony Hammond

Examples

```
# Make up Some data:
Q <- c(177.685, 240.898, 221.954, 205.55, 383.051, 154.061, 216.582)
Stage <- c(1.855, 2.109, 2.037, 1.972, 2.574, 1.748, 2.016)
Observations <- data.frame(Q, Stage)
#apply the rating function:
Rating(Observations)
#Hold the stage correction at zero
Rating(Observations, a = 0)
```

ReFH *Revitalised Flood Hydrograph Model (ReFH)*

Description

Provides outputs of the ReFH model from catchment descriptors or user defined inputs

Usage

```
ReFH(
  CDs = NULL,
  Depth = NULL,
  duration = NULL,
  timestep = NULL,
  scaled = NULL,
  PlotTitle = NULL,
  RPa = NULL,
  alpha = TRUE,
  season = NULL,
  AREA = NULL,
  TP = NULL,
  BR = NULL,
  BL = NULL,
  Cmax = NULL,
  Cini = NULL,
  BFini = NULL,
  Rain = NULL
)
```

Arguments

CDs	catchment descriptors derived from either GetCDs or ImportCD
Depth	a numeric value. The depth of rainfall used as input in the estimation of a design hydrograph. The default, when Depth = NULL, is a two year rainfall.
duration	a numeric value. A duration for the design rainfall
timestep	a numeric value. A user defined data interval. The default changes depending on the estimated time to peak to formulate a sensible looking result
scaled	a numeric value of peak flow in m3/s
PlotTitle	a character string. A user defined title for the ReFH plot
RPa	return period for alpha adjustment. This is only for the purposes of the alpha adjustment, it doesn't change the rainfall input
alpha	a logical argument with default TRUE. If TRUE the alpha adjustment is applied based on RPa. If FALSE, no alpha adjustment is made
season	a choice of "summer" or "winter". The default is "summer" in urban catchments (URBEXT2000 > 0.03) and "winter" in rural catchments

AREA	numeric. Catchment area in km2.
TP	numeric. Time to peak parameter (hours)
BR	numeric. Baseflow recharge parameter
BL	numeric. Baseflow lag parameter (hours)
Cmax	numeric. Maximum soil moisture capacity parameter (mm)
Cini	numeric. Initial soil moisture content (mm)
BFini	numeric. Initial baseflow (m3/s)
Rain	numeric. User input rainfall. A numeric vector

Details

The ReFH is described in the Flood Estimation Handbook Supplementary Report No.1 (2007). The method to derive design rainfall profiles is described in the Flood Estimation Handbook (1999), volume 2. Users can also input their own rainfall with the 'Rain' argument. As a default, when catchment descriptors (CDs) are provided the ReFH function uses catchment descriptors to estimate the parameters of the ReFH model and the two year rainfall for the critical duration. The latter is based on a quadratic interpolation of the catchment descriptors RMED1H, RMED1D, and RMED2D. Parameters and initial conditions can also be individually input by the user. If a parameter argument is used for one or more of the parameters, then these overwrite the CD derived parameters. If a value for the scaled argument is provided (m3/s), a scaled hydrograph is returned. The RPa argument doesn't change the rainfall input and is only needed for the alpha adjustment (see the FEH supplement report no.1).

Value

A print out of parameters, a results data.frame, and a plot. First is a print of the parameters, initial conditions and the catchment area. The second is a data.frame with columns Rain, NetRain, Runoff, Baseflow, and TotalFlow. If the scale argument is used a numeric vector containing the scaled hydrograph is returned instead of the results dataframe. The plot is of the ReFH output, with rainfall, net-rainfall, baseflow, runoff and total flow. If the scaled argument is used, a scaled hydrograph is plotted.

Author(s)

Anthony Hammond

Examples

```
#Get CDs and apply the ReFH function
CDs.203018 <- GetCDs(203018)
ReFH(CDs.203018)
#Apply the ReFH function, scale to a 100-year flow estimate and change the plot title accordingly
ReFH(CDs.203018, scaled = 182, PlotTitle = "100-Year Design Hydrograph - Site 203018")
#Apply the ReFH function with a user defined initial baseflow
ReFH(CDs.203018, BFini = 6)
```

SCF *Seasonal correction factor (SCF)*

Description

The results of applying the ratio of the seasonal annual maximum rainfall for a given duration to the annual maximum rainfall for the same duration

Usage

```
SCF(SAAR, duration)
```

Arguments

SAAR	standardised average annual rainfall. Numeric
duration	duration in hours. Numeric

Details

The SCF and it's use is detailed in R&D Technical Report FD1913/TR - Revitalisation of the FSR/FEH rainfall runoff method (2005). The ReFH model has a design rainfall profile included for winter and summer but the depth duration frequency (DDF) model is calibrated on annual maximum peaks as opposed to seasonal peaks. The SCF is necessary to convert the DDF estimate to a seasonal one. Similarly, the DDF model is calibrated on point rainfall and the area reduction factor converts it to a catchment rainfall for use with a rainfall runoff model such as ReFH (see details of the ReFH function).The final depth, therefore is; Depth = DDFdepth x ARF x SCF.

Value

A data.frame of one row and two columns: SCFSummer and SCFWinter.

Author(s)

Anthony Hammond

Examples

```
#Derive the SCFs for a SAAR of 1981 and a duration of 6.5
SCF(1981, 6.5)
```

SimData *Data simulator*

Description

Simulation of a random sample from the generalised extreme value, generalised logistic, Gumbel, or generalised Pareto distributions

Usage

```
SimData(n, pars = NULL, dist = "GenLog", GF = NULL)
```

Arguments

n	sample size to be simulated
pars	vector of parameters in the order of location, scale, shape (only location and shape for Gumbel)
dist	choice of distribution. Either "GEV", "GenLog", "Gumbel" or "GenPareto"
GF	vector of GF inputs in the order of Lcv, LSkew, QMED (only Lcv and QMED if dist = "Gumbel")

Details

The simulated sample can be generated using distribution parameters, or the growth factor (GF) inputs; linear coefficient of variationn (Lcv), linear skewness (LSkew) & the median annual maximum (QMED).

Value

A random sample of size n for the chosen distribution.

Author(s)

Anthony Hammond

Examples

```
#Simulate a sample of size 30 using parameters GenLog and parameters 299, 51, -0.042
SimData(30, pars = c(299, 51, -0.042), dist = "GenLog")
#Now simulate using the Lcv, Lskew, and median (0.17, 0.04, 310)
SimData(30, GF = c(0.17, 0.04, 310), dist = "GenLog")
```

`ThamesPQ` *Kingston upon Thames daily flow and catchment precipitation 2000-10-01 to 2015-09-30*

Description

A data.frame of three columns; Date, Precipitation (P), & daily mean flow (Q)

Usage

```
ThamesPQ
```

Format

A data frame with 5478 rows and 3 columns:

Date Date

P Precipitation, in mm

Q Daily mean discharge, in m3/s

Source

https://nrfa.ceh.ac.uk/data/station/meanflow/39001

`TrendTest` *Trend hypothesis test*

Description

A hypothesis test for the correlation between the variable of interest and time

Usage

```
TrendTest(x, method = "kendall")
```

Arguments

`x`	a numeric vector or a data.frame with dates in the first column and chronologically ordered variable in the second.
`method`	a choice of test method. Choices are "pearson", "spearman", and "kendall"

Details

The test can be performed on a numeric vector, or a data.frame with dates in the first column and the associated variable of interest in the second. A choice can be made between a Pearson's, Spearman's Rho or Kendall's tau test. The Spearman and Kendall are based on ranks and will therefore have the same results whether dates are included or not. The default is kendall (note: for very long time series the kendall method takes a touch longer).

Value

A data.frame with columns and associated values: P_value, correlation coefficient, and method specific statistic.

Author(s)

Anthony Hammond

Examples

```
#Get AMAX sample and apply a trend test with the default kendall test.
AM.27083 <- GetAM(27083)
TrendTest(AM.27083)
#Apply the test with the pearson method with dates included and not
TrendTest(AM.27083, method = "pearson")
TrendTest(AM.27083$Flow, method = "pearson")
```

`UAF` *Urban adjustment factor (UAF) and percentage runoff urban adjustment factor (PRUAF)*

Description

UAF and PRUAF from catchment descriptors for QMED estimation in ungauged urban catchments

Usage

```
UAF(CDs = NULL, URBEXT2000, BFIHOST)
```

Arguments

`CDs`	catchment descriptors derived from either GetCDs or ImportCDs
`URBEXT2000`	quantification of catchment urbanisation (used when CDs is not)
`BFIHOST`	baseflow index as a function of hydrological soil type of the catchment (used when CDs is not)

Value

a data.frame with columns UAF and PRUAF

Author(s)

Anthony Hammond

Examples

```
#Get some catchment descriptors for an urban catchment calculate the UAF & PRUAF
CDs.53006 <- GetCDs(53006)
UAF(CDs.53006)
#Calculate UAF and PRUAF using a user input URBEXT2000 and BFIHOST
UAF(URBEXT2000 = 0.1138, BFIHOST = 0.3620)
```

`UKOutline` *UK outline*

Description

Easting and northing national grid reference points around the coast of the UK

Usage

```
UKOutline
```

Format

A data frame with 3867 rows and 2 variables

X_BNG Easting, British national grid reference

Y_BNG Northing, British national grid reference

Source

https://environment.data.gov.uk/

`Uncertainty` *Uncertainty quantification for gauged and ungauged pooled estimates*

Description

Quantification of uncertainty for pooling results for the gauged and ungauged case

Usage

```
Uncertainty(
  x,
  gauged = FALSE,
  RP = 100,
  dist = "GenLog",
  qmed = NULL,
  no.Donors = 2,
  UrbAdj = FALSE,
  CDs = NULL,
  conf = 0.95
)
```

Arguments

`x`	the pooled group derived from the Pool() function
`gauged`	a logical argument with a default of FALSE. If FALSE the uncertainty intervals are calculated for the ungauged case. If TRUE they are calculated for the gauged case
`RP`	the return period of interest. Default is 100
`dist`	a choice of distribution to use for the estimates. Choices are "GEV", "GenLog" or "Gumbel". The default is "GenLog"
`qmed`	the QMED estimate for the ungauged case. Or for the gauged if the user wishes to override the median from the NRFA data
`no.Donors`	number of donors used for estimation of QMED in the ungauged case
`UrbAdj`	applies an urban adjustment to the growth curves
`CDs`	catchment descriptors derived from either GetCDs or ImportCDs. Necessary if a UrbAdj is TRUE
`conf`	the confidence level of the intervals for the gauged case. Default is 0.95. Must be between 0 and 1

Details

Uncertainty in the ungauged case is calulated as equations 8, 9, and 10 in Hammond, A. (2021). Sampling uncertainty of UK design flood estimation. Hydrology Research, 52 (6), 1357–1371. The 68 percent and 95 percent intervals are returned. For the gauged case the pooled group is bootstrapped 500 times and the enhanced single site weighted linear skewness (LSkew) and linear coefficient of variation (Lcv) are calculated 500 times accordingly and 500 associated growth factors are calculated. Each growth factor (GF) is multiplied by a randomly selected median annual maximum flow (QMED) from the uncertainty distribution of median estimates for the gauged subject site. The distribution of medians is derived from bootstrapping the gauged site 500 times. The intervals are then the upper and lower quantiles (depending on the conf input) of the distribution of median * GFs. For the gauged case the user can choose the level for the intervals. The default is 0.95. Occasionally the single site central estimate will be outside the uncertainty intervals. In these cases the intervals are widened to incorporate it. i.e. if above the intervals, the upper interval is increased to the single site estimate and vice versa if below. This occurs regardless of the confidence setting. For details about the calculations of weighted growth curves & urban adjustment see the PoolEst() function details. A trend option is not included within the Uncertainty function and would need be considered separately if used in PoolEst. An indication of the uncertainty for trend applied in PoolEst is provided in the PoolEst function details. The gauged method is also detailed in Hammond (2021).

Value

For the ungauged case a data.frame of four values relating to the lower 68 and upper 68 percent interval and the lower 95 and upper 95 percent intervals. These are headed by the associated percentiles. For the gauged case a numeric vector of two values is provided with the lower and upper intervals of the chosen conf level. The uncertainty function doesn't have a trend option; if trend is used in the pooled estimate this would need to be considered and intervals adjused accordingly. However a greater uncertainty should be considered.

Author(s)

Anthony Hammond

Examples

```
#Get CDs, form an ungauged pooling group and quantify the uncertainty of the
#50-year pooled estimate when using a CDs estimate of QMED with no donors
CDs.203018 <- GetCDs(203018)
Pool.203018 <- Pool(CDs.203018, exclude = 203018)
Uncertainty(Pool.203018, qmed  = QMED(CDs.203018), no.Donors = 0, RP = 50)
#Form pooling group with subject site included. Quantify the uncertainty of the
#50-year pooled estimate at the 99% level.
 Pool.203018 <- Pool(CDs.203018)
 Uncertainty(Pool.203018, gauged = TRUE, RP = 50, conf = 0.99)
```

UncSS *Uncertainty for the single site*

Description

Quantifies the aleatoric uncertainty for a single site estimate, by bootstrapping the sample

Usage

```
UncSS(x, func, conf = 0.95, RP = FALSE)
```

Arguments

x	a numeric vector. The sample of interest
func	the function to be applied
conf	the confidence level of the intervals
RP	return period. Necessary if func requires RP

Details

The bootstrapping procedure resamples from a sample N*500 times with replacement. After splitting into 500 samples of size N, the statsitic of interest is calculated on each. upper and lower quantiles of the resulting distribution are used as the quantification of uncertainty. Any function that provides an estimate based on a sample of data can be used. Including any function that provides estimates as a function of return period.

Value

A data.frame of three values; central, lower, and upper bootstrapped estimates.

Author(s)

Anthony Hammond

Examples

```
#Extract an AMAX sample and quantify uncertainty for the GEV estimated 50-year flow
AM.203018 <- GetAM(203018)
UncSS(AM.203018$Flow, func = GEVAM, RP = 50)
#Quantify uncertainty for the sample standard deviation at the 90 percent confidence level
UncSS(AM.203018$Flow, func = sd, conf = 0.90)
```

WeightsGLcv *Site gauged linear coefficient of variation (Lcv) weightings*

Description

Provides the gauged Lcv weights for each site in a pooling group

Usage

```
WeightsGLcv(x)
```

Arguments

x	pooling group derived with the Pool() function

Details

Weighting method as according to Science Report: SC050050 - Improving the FEH statistical procedures for flood frequency estimation

Value

A data.frame with site references in the first column and associated weights in the second

Author(s)

Anthony Hammond

Examples

```
#Get some CDs, form a gauged pooling group, and estimate gauged Lcv
CDs.96001 <- GetCDs(96001)
Pool.96001 <- Pool(CDs.96001)
WeightsGLcv(Pool.96001)
```

WeightsGLSkew — *Site gauged linear skewness (LSkew) weightings*

Description

Provides the gauged LSkew weights for each site in a pooling group

Usage

```
WeightsGLSkew(x)
```

Arguments

x	pooling group derived with the Pool() function

Details

Weighting method as according to Science Report: SC050050 - Improving the FEH statistical procedures for flood frequency estimation

Value

A data.frame with site references in the first column and associated weights in the second

Author(s)

Anthony Hammond

Examples

```
#Get some CDs, form a gauged pooling group, and estimate gauged LSkew
CDs.96001 <- GetCDs(96001)
Pool.96001 <- Pool(CDs.96001)
WeightsGLSkew(Pool.96001)
```

WeightsUnLcv *Site ungauged linear coefficient of variation (Lcv) weightings*

Description

Provides the ungauged Lcv weights for each site in a pooling group

Usage

```
WeightsUnLcv(x)
```

Arguments

x	pooling group derived with the Pool() function

Details

Weighting method as according to Science Report: SC050050 - Improving the FEH statistical procedures for flood frequency estimation

Value

A data.frame with site references in the first column and associated weights in the second

Author(s)

Anthony Hammond

Examples

```
#Get some CDs, form an ungauged pooling group, and estimate ungauged Lcv
CDs.96001 <- GetCDs(96001)
Pool.96001 <- Pool(CDs.96001, exclude = 96001)
WeightsUnLcv(Pool.96001)
```

WeightsUnLSkew *Site ungauged linear skewness (LSkew) weightings*

Description

Provides the ungauged LSkew weights for each site in a pooling group

Usage

```
WeightsUnLSkew(x)
```

Arguments

x pooling group derived with the Pool() function

Details

Weighting method as according to Science Report: SC050050 - Improving the FEH statistical procedures for flood frequency estimation

Value

A data.frame with site references in the first column and associated weights in the second

Author(s)

Anthony Hammond

Examples

```
#Get some CDs, form an ungauged pooling group, and estimate ungauged LSkew
CDs.96001 <- GetCDs(96001)
Pool.96001 <- Pool(CDs.96001, exclude = 96001)
WeightsUnLSkew(Pool.96001)
```

WGaugLcv *Gauged pool weighted linear coefficient of variation (Lcv)*

Description

Calculates the gauged weighted Lcv from a pooling group (enhanced single site)

Usage

```
WGaugLcv(x)
```

Arguments

x pooling group derived with the Pool() function

Details

Weighting method as according to Science Report: SC050050 - Improving the FEH statistical procedures for flood frequency estimation

Value

the gauged weighted Lcv from a pooling group

Author(s)

Anthony Hammond

Examples

```
#Get some CDs, form a gauged pooling group, and estimate gauged Lcv
CDs.96001 <- GetCDs(96001)
Pool.96001 <- Pool(CDs.96001)
WGaugLcv(Pool.96001)
```

`WGaugLSkew` *Gauged pool weighted linear skewness (LSkew)*

Description

Calculates the gauged weighted LSkew from a pooling group (enhanced single site)

Usage

```
WGaugLSkew(x)
```

Arguments

x	pooling group derived with the Pool() function

Details

Weighting method as according to Science Report: SC050050 - Improving the FEH statistical procedures for flood frequency estimation

Value

the gauged weighted LSkew from a pooling group

Author(s)

Anthony Hammond

Examples

```
#Get some CDs, form a gauged pooling group, and estimate gauged LSkew
CDs.96001 <- GetCDs(96001)
Pool.96001 <- Pool(CDs.96001)
WGaugLSkew(Pool.96001)
```

WungLcv *Ungauged pool weighted linear coefficient of variation (Lcv)*

Description

Calculates the ungauged weighted Lcv from a pooling group

Usage

```
WungLcv(x)
```

Arguments

x	pooling group derived with the Pool() function

Details

Weighting method as according to Science Report: SC050050 - Improving the FEH statistical procedures for flood frequency estimation

Value

the ungauged weighted Lcv from a pooling group

Author(s)

Anthony Hammond

Examples

```
#Get some CDs, form an ungauged pooling group, and estimate ungauged Lcv
CDs.96001 <- GetCDs(96001)
Pool.96001 <- Pool(CDs.96001, exclude = 96001)
WungLcv(Pool.96001)
```

WungLSkew *Ungauged pool weighted linear skewness (LSkew)*

Description

Calculates the ungauged weighted LSkew from a pooling group

Usage

```
WungLSkew(x)
```

Arguments

x pooling group derived with the Pool() function

Details

Weighting method as according to Science Report: SC050050 - Improving the FEH statistical procedures for flood frequency estimation

Value

the ungauged weighted LSkew from a pooling group

Author(s)

Anthony Hammond

Examples

```
#Get some CDs, form an ungauged pooling group, and estimate ungauged LSkew
CDs.96001 <- GetCDs(96001)
Pool.96001 <- Pool(CDs.96001, exclude = 96001)
WungLSkew(Pool.96001)
```

Zdists *Zdist Goodness of fit measure for pooling groups*

Description

Calculates the goodness of fit score for pooling groups with the method outlined in the Flood Estimation Hanbook (1999), volume 3.

Usage

```
Zdists(x)
```

Arguments

x pooling group derived from the Pool() function

Details

The goodness of fit measure was developed by Hosking & Wallis and can be found in their book ’Regional Frequency Analysis: an approach based on LMoments (1997), as well as Flood Estimation Handbook volume 3.

Value

A list with the first element a data.frame of three GoF scores related to the columns; "GEV, "GenLog", and "Gumebl. The second element is a character stating which has the best fit.

Author(s)

Anthony Hammond

Examples

```
#Get CDs, form a pooling group and calculate the Zdist
CDs.203018 <- GetCDs(203018)
Pool.203018 <- Pool(CDs.203018)
Zdists(Pool.203018)
```

Index