Package ‘nsRFA’

October 13, 2022

Version 0.7-15
Date 2020-02-25
Title Non-Supervised Regional Frequency Analysis
Depends R (>= 3.0.0), stats, graphics
Description A collection of statistical tools for objective (non-supervised) applications of the Regional Frequency Analysis methods in hydrology. The package refers to the index-value method and, more precisely, helps the hydrologist to: (1) regionalize the index-value; (2) form homogeneous regions with similar growth curves; (3) fit distribution functions to the empirical regional growth curves. Most of the methods are those described in the Flood Estimation Handbook (Centre for Ecology & Hydrology, 1999, ISBN:9781906698003). Homogeneity tests from Hosking and Wallis (1993) <doi:10.1029/92WR01980> and Viglione et al. (2007) <doi:10.1029/2006WR005095> are available.
License GPL (>= 2)
Copyright The fortran routines for the function HW.original are from Jonathan R. M. Hosking and are available at the url http://www.research.ibm.com/people/h/hosking/lmoments.html. The functions ‘MLlaio2004.R’, ‘GOFlaio2004.R’ and ‘MSCliao2008.R’ are derived from original Francesco Laio Matlab code (non published material). The fortran routines for the function ‘bestlm’ are from Alan Miller and were obtained from his R package ‘leaps’.
NeedsCompilation yes
Author Alberto Viglione [aut, cre],
Jonathan R. M. Hosking [ctb],
Francesco Laio [ctb],
Alan Miller [ctb],
Eric Gaume [ctb],
Olivier Payrastre [ctb],
Jose Luis Salinas [ctb],
Chi Cong N’guyen [ctb],
Karine Halbert [ctb]
Maintainer Alberto Viglione <alberto.viglione@polito.it>
R topics documented:

- nsRFA-package
- AD.dist
- Ardechedata
- BayesianMCMC
- bestlm
- DIAGNOSTICS
- DISTPLOTS
- EXP
- FEH1000
- functionsLaio
- GENLOGIS
- GENPAR
- GEV
- GOFlaio2004
- GOFmontecarlo
- GUMBEL
- HOMTESTS
- HW.original
- hydroSIMN
- KAPPA
- Lmoments
- LOGNORM
- MLlaio2004
- moments
- MSClaio2008
- P3
- REGRDIAGNOSTICS
- roi
- SERIESPLOTS
- STATICPLOTS
- traceWminim
- varLmoments

Index
Non-supervised Regional Frequency Analysis

Description

The estimation of hydrological variables in ungauged basins is a very important topic for many purposes, from research to engineering applications and water management (see the PUB project, Sivapalan et al., 2003). Regardless of the method used to perform such estimation, the underlying idea is to transfer the hydrological information from gauged to ungauged sites. When observations of the same variable at different measuring sites are available and many data samples are used together as source of information, the methods are called regional methods. The well known regional frequency analysis (e.g. Hosking and Wallis, 1997), where the interest is in the assessment of the frequency of hydrological events, belong to this class of methods. In literature, the main studied variable is the flood peak and the most used regional approach is the index-flood method of Dalrymple (1960), in which it is implicitly assumed that the flood frequency distribution for different sites belonging to a homogenous region is the same except for a site-specific scale factor, the index-flood (see Hosking and Wallis, 1997, for details). Hence, the estimation of the flood frequency distribution for an ungauged site can be divided into two parts: estimation of the index-flood (more in general, the index-value) through linear/non-linear relations with climatic and basin descriptors; estimation of the adimentional flood frequency distribution, the growth curve, assigning the ungauged basin to one homogeneous region.

nsRFA is a collection of statistical tools for objective (non-supervised) applications of the Regional Frequency Analysis methods in hydrology. This does not mean that Regional Frequency Analysis should be non-supervised. These tools are addressed to experts, to help their expert judgement. The functions in nsRFA allow the application of the index-flood method in the following points:

- regionalization of the index-value;
- formation of homogeneous regions for the growth curves;
- fit of a distribution function to the empirical growth curve of each region;

Regionalization of the index-value

The index-value can be either the sample mean (e.g. Hosking and Wallis, 1997) or the sample median (e.g. Robson and Reed, 1999) or another scale parameter. Many methodological approaches are available for the index-value estimation, and their differences can be related to the amount of information available. Excluding direct methods, that use information provided by flow data available at the station of interest, regional estimation methods require ancillary hydrological and physical information. Those methods can be divided in two classes: the multiregressive approach and the hydrological simulation approach. For both of them, the best estimator is the one that optimizes some criterion, such as the minimum error, the minimum variance or the maximum efficiency. Due to its simplicity, the most frequently used method is the multiregressive approach (see e.g. Kottegoda & Rosso, 1998; Viglione et al., 2007a), that relates the index-flood to catchment characteristics, such as climatic indices, geologic and morphologic parameters, land cover type, etc., through linear or non-linear equations.

R provides the functions \texttt{lm} and \texttt{nls} for linear and non-linear regressions (package \texttt{stats}). With
the package nsRFA, a tool to select the best linear regressions given a set of candidate descriptors, bestlm, is provided. In REGRDIAGNOSTICS several functions are provided to analyze the output of lm, such as: the coefficient of determination (classic and adjusted); the Student t significance test; the variance inflation factor (VIF); the root mean squared error (RMSE); the mean absolute error (MAE); the prediction intervals; predicted values by a jackknife (cross-validation) procedure. The functions in DIAGNOSTICS provide more general diagnostics of model results (that can be also non-linear), comparing estimated values with observed values.

More details are provided in vignettes:

```r
nsRFA_ex01  How to use the package nsRFA (example 1):
             Regional frequency analysis of the annual flows in Piemonte
             and Valle d’Aosta
```

that can be accessed via `vignette("nsRFA_ex01", package="nsRFA")`.

### Formation of homogeneous regions for the growth curves

Different techniques exist, for example those that lead to the formation of fixed regions through cluster analysis (Hosking and Wallis, 1997, Viglione, 2007), or those based on the method of the region of influence (ROI, Burn, 1990). The regional procedure can be divided into two parts: the formation of regions and the assignment of an ungauged site to one of them. Concerning the first point, the sites are grouped according to their similarity in terms of those basin descriptors that are assumed to explain the shape of the growth curve. This shape is usually quantified in a parametric way. For instance, the coefficient of variation (CV) or the L-CV of the curve can be used for this purpose. The package nsRFA provides the functions in moments and Lmoments to calculate sample moments and L-moments. Subsequently, the selected parameter is related with basin descriptors through a linear or a more complex model. A regression analysis is performed with different combination of descriptors, and descriptors that are strongly related with the parameter are used to group sites in regions. The same tools used for the regionalization of the index value, i.e. bestlm, REGRDIAGNOSTICS and DIAGNOSTICS, can be used if the parametric method is chosen.

nsRFA also provide a non-parametric approach that considers the dimensionless growth curve as a whole (see, Viglione et al., 2006; Viglione, 2007). The multiregressive approach can still be used if we reason in terms of (dis)similarity between pairs of basins in the following way: (1) for each couple of stations, a dissimilarity index between non-dimensional curves is calculated using a quantitatively predefined metric, for example using the Anderson-Darling statistic (A2), and organising the distances in a matrix with `AD.dist`; (2) for each basin descriptor, the absolute value (or another metric) of the difference between its measure in two basins is used as distance between them, using `dist` of the package `stats` to obtain distance matrices; (4) a multiregressive approach (`bestlm, lm`) is applied considering the matrices as variables and the basin descriptors associated to the best regression are chosen; (5) the significance of the linear relationship between distance matrices is assessed through the Mantel test with `mantel.lm`.

In the suitable-descriptor’s space, stations with similar descriptor values can be grouped into disjoint regions through a cluster analysis (using functions in `traceWminim`) or the ROI method can be used adapting a region to the ungauged basin (`roi`). In both cases, the homogeneity of the regions can be assessed with the functions in HOMTESTS, where the Hosking and Wallis heterogeneity measures (HW.tests, see Hosking and Wallis, 1997) and the Anderson-Darling homogeneity test (ADbootstrap.test, see Viglione et al., 2007b) are provided.

More details are provided in vignettes:
How to use the package nsRFA (example 1):
Regional frequency analysis of the annual flows in Piemonte and Valle d’Aosta

How to use the package nsRFA (example 2):
Region-Of-Influence approach, some FEH examples

that can be accessed via vignette("nsRFA_ex01", package="nsRFA").

**Fit of a distribution function to the empirical growth curve of each region**

Once an homogeneous region is defined, the empirical growth curves can be pooled together and a probability distribution can be fitted to the pooled sample. The choice of the best distribution can be assisted by a Model Selection Criteria with **MSClaio2008** (see, Laio et al., 2008). The parameters of the selected distribution can be estimated using the method of moments (**moment_estimation**), L-moments (**par.GEV, par.genpar, par.gamma**, ...) or maximum-likelihood (**MLlaio2004**). Goodness-of-fit tests are also available: the Anderson-Darling goodness of fit test with **GOFlaio2004** (Laio, 2004), and Monte-Carlo based tests with **GOFmontecarlo**. Confidence intervals for the fitted distribution can be calculated with a Markov Chain Monte Carlo algorithm, using **BayesianMCMC**.

More details are provided in vignettes:

- **nsRFA_ex01** How to use the package nsRFA (example 1):
  Regional frequency analysis of the annual flows in Piemonte and Valle d’Aosta

- **MSClaio2008** Model selection techniques for the frequency analysis of hydrological extremes: the MSClaio2008 R function

that can be accessed via vignette("nsRFA_ex01", package="nsRFA").

**Other functions**

**varLmoments** provides distribution-free unbiased estimators of the variances and covariances of sample L-moments, as described in Elamir and Seheult (2004).

More details are provided in vignettes:

- **Fig1ElamirSeheult** Figure 1 in Elamir and Seheult (2004)

**Details**

- **Package**: nsRFA
- **Version**: 0.7

The package provides several tools for Regional Frequency Analysis of hydrological variables. The first version dates to 2006 and was developed in Turin at the Politecnico by Alberto Viglione.

For a complete list of the functions, use `library(help="nsRFA")`.

**Main changes in version 0.7**
Main changes in version 0.6

0.6-9: new vignette Fig11GriffisStedinger;
0.6-8: exponential and Gumbel distributions added in GOFmontecarlo;
0.6-6: some plotting position/probability plots have been added in DISTPLOTS;
0.6-4: refinement of function BayesianMCMC;
0.6-2: new vignette nsRFA_ex02;
0.6-2: refinement of function BayesianMCMC;
0.6-0: new vignette nsRFA_ex01;
0.6-0: new function bestlm;
0.6-0: the plotting position/probability plots in DISTPLOTS have been reshaped;
0.6-0: this list of changes has been added;

Author(s)

Alberto Viglione
Maintainer: Alberto Viglione <viglione@hydro.tuwien.ac.at>

References

All the manual references are listed here:


---

**AD.dist**

*Anderson-Darling distance matrix for growth curves*

**Description**

Distance matrix for growth curves. Every element of the matrix is the Anderson-Darling statistic calculated between two series.

**Usage**

`AD.dist (x, cod, index=2)`

**Arguments**

- `x`: vector representing data from many samples defined with `cod`
- `cod`: array that defines the data subdivision among sites
- `index`: if `index`=1 samples are divided by their average value; if `index`=2 (default) samples are divided by their median value

**Details**

The Anderson-Darling statistic used here is the one defined in [http://en.wikipedia.org/wiki/Anderson-Darling_test](http://en.wikipedia.org/wiki/Anderson-Darling_test) as $A^2$. 
**Value**

$AD_{dist}$ returns the distance matrix between growth-curves built with the Anderson-Darling statistic.

**Note**

For information on the package and the Author, and for all the references, see nsRFA.

**See Also**

traceWminim, roi.

**Examples**

```r
data(hydroSIMN)
annualflows
summary(annualflows)
x <- annualflows["dato"],
cod <- annualflows["cod",

# Ad.dist
AD.dist(x,cod) # it takes some time
```

**Description**


**Usage**

```r
data(Ardechedata)
```
Format

Ardeche_areas, areas (km²) of the gauged and ungauged catchments in the Ardèche region (France); Ardeche_ungauged_extremes, flood peaks (m³/s) reconstructed in ungauged catchments and number of years for which the peak was not exceeded; Beauvene_cont, systematic flood peaks (m³/s) recorded at one station; Chambonas_cont, systematic flood peaks (m³/s) recorded at one station; SaintLaurent_cont, systematic flood peaks (m³/s) recorded at one station; SaintMartin_cont, systematic flood peaks (m³/s) recorded at one station; SaintMartin_hist, values for historical peaks (m³/s) for one station and for flood perception thresholds (m³/s) non exceeded in the periods indicated; Vogue_cont, systematic flood peaks (m³/s) recorded at one station.

Examples

data(Ardechedata)
SaintMartin_cont
SaintMartin_hist

BayesianMCMC

Bayesian MCMC frequency analysis

Description

Bayesian Markov Chain Monte Carlo algorithm for flood frequency analysis with historical and other information. The user can choose between a local and a regional analysis.

Usage

BayesianMCMC (xcont, xhist=NA, infhist=NA, suphist=NA, nbans=NA, seuil=NA, nbpas=1000, nbchaines=3, confint=c(0.05, 0.95), dist="GEV", apriori=function(...){1}, parameters0=NA, varparameters0=NA)

BayesianMCMCcont (x, nbpas=NA)

BayesianMCMCreg (xcont, scont, xhist=NA, infhist=NA, suphist=NA, shist=NA, nbans=NA, seuil=NA, nbpas=1000, nbchaines=3, confint=c(0.05, 0.95), dist="GEV", apriori=function(...){1}, parameters0=NA, varparameters0=NA)

BayesianMCMCregcont (x, nbpas=NA)

plotBayesianMCMCreg_surf (x, surf, ask=FALSE, ...)

# S3 method for class 'BayesianMCMC'
plot(x, which=1, ask=FALSE, ...)
# S3 method for class 'BayesianMCMC'
print(x, ...)
# S3 method for class 'BayesianMCMC'
plot(x, which=1, ask=FALSE, ...)
# S3 method for class 'BayesianMCMC'
print(x, ...)
Arguments

- **x**: object of class `BayesianMCMC`, output of function `BayesianMCMC`
- **xcont**: vector of systematic data
- **scont**: vector of upstream catchment surfaces of systematic data
- **xhist**: vector of historical data and/or extreme discharges at ungauged sites
- **infhist**: vector of inferior limit for historical data and/or extreme discharges at ungauged sites
- **suphist**: vector of superior limit for historical data and/or extreme discharges at ungauged sites
- **shist**: vector of upstream catchment surfaces of extreme discharges at ungauged sites
- **nbans**: period (in years) over which every threshold has not been exceeded except for the historical data and/or extreme discharges at ungauged sites. If several values of xhist for a same threshold, put the number of years associated to the threshold on the first row, then put 0 (see examples)
- **seuil**: threshold not exceeded in the historical period except for the historical data and/or extreme discharges at ungauged sites (several thresholds allowed).
- **nbpas**: number of iterations for the MCMC algorithm
- **nbchaines**: number of chains for the MCMC algorithm
- **confint**: confidence limits for the flood quantiles
- **apriori**: function of the parameters of the model ‘proportional to’ their a-priori guessed distribution. The default function returns always 1, i.e. there is no a-priori information
- **parameters0**: initial values of the parameters for the MCMC algorithm
- **varparameters0**: initial values of the parameter variances for the MCMC algorithm
- **which**: a number of a vector of numbers that defines the graph to plot (see details)
- **ask**: if TRUE, the interactive mode is run
- **surf**: a particular surface (number or vector), not necessarily being a surface included in the scont or shist vectors
- **...**: other arguments

Details

Supported cases

- Using only the systematic data: xcont provided, xhist=NA, infhist=NA and suphist=NA
- Using censored information, historical flood known: xcont and xhist provided, infhist=NA and suphist=NA
- Using censored information, historical flood unknown precisely but its lower limit known: xcont and infhist provided, xhist=NA and suphist=NA
- Taking into account flood estimation intervals: infhist
and suphist (respectively lower and upper limits) provided, xcont provided, xhist=NA - Please note that every other case is NOT supported. For example, you can’t have some historical flood values perfectly known as well as some other for which you only know a lower limit or an interval.

Regarding the perception thresholds: - By definition, the number of exceedances of each perception threshold within its application period has to be known precisely, and all the floods exceeding the threshold have to be included in xhist (or infhist or suphist). - Several thresholds are allowed. - It is possible to include in xhist (or infhist or suphist) historical values that do not exceed the associated perception threshold. - If for one or several thresholds you only know that this or these threshold have never been exceeded and no more information is available on floods that did not exceed the threshold(s), this case is also supported. In this case, put for the historical flood corresponding to the threshold xhist=-1 (or infhist=-1 or infhist=suphist=-1).

Bayesian inference

Bayesian inference uses a numerical estimate of the degree of belief in a hypothesis before evidence has been observed and calculates a numerical estimate of the degree of belief in the hypothesis after evidence has been observed. The name ‘Bayesian’ comes from the frequent use of Bayes’ theorem in the inference process. In our case the problem is: which is the probability that a frequency function \( P \) (of type defined in \( \text{dist} \)) has parameters \( \theta \), given that we have observed the realizations \( D \) (defined in \( \text{xcont, xhist, infhist, suphist, nbans, seuil} \)). The Bayes’ theorem writes

\[
P(\theta|D) = \frac{P(D|\theta) \cdot P(\theta)}{P(D)}
\]

where \( P(\theta|D) \) is the conditional probability of \( \theta \), given \( D \) (it is also called the posterior probability because it is derived from or depends upon the specified value of \( D \)) and is the result we are interested in; \( P(\theta) \) is the prior probability or marginal probability of \( \theta \) (‘prior’ in the sense that it does not take into account any information about \( D \)), and can be given using the input apriori (it can be used to account for causal information); \( P(D|\theta) \) is the conditional probability of \( D \) given \( \theta \) and it is defined choosing \( \text{dist} \) and depending on the availability of historical data; \( P(D) \) is the prior or marginal probability of \( D \), and acts as a normalizing constant. Intuitively, Bayes’ theorem in this form describes the way in which one’s beliefs about observing \( \theta \) are updated by having observed \( D \).

Since complex models cannot be processed in closed form by a Bayesian analysis, namely because of the extreme difficulty in computing the normalization factor \( P(D) \), simulation-based Monte Carlo techniques as the MCMC approaches are used.

MCMC Metropolis algorithm

Markov chain Monte Carlo (MCMC) methods (which include random walk Monte Carlo methods), are a class of algorithms for sampling from probability distributions based on constructing a Markov chain that has the desired distribution as its equilibrium distribution. The state of the chain after a large number of steps is then used as a sample from the desired distribution. The quality of the sample improves as a function of the number of steps.

The MCMC is performed here through a simple Metropolis algorithm, i.e. a Metropolis-Hastings algorithm with symmetric proposal density. The Metropolis-Hastings algorithm can draw samples from any probability distribution \( P(x) \), requiring only that a function proportional to the density can be calculated at \( x \). In Bayesian applications, the normalization factor is often extremely difficult to compute, so the ability to generate a sample without knowing this constant of proportionality is a major virtue of the algorithm. The algorithm generates a Markov chain in which each state \( x_{t+1} \) depends only on the previous state \( x_t \). The algorithm uses a Gaussian proposal density \( N(x_t, \sigma_x) \),
which depends on the current state $x_t$, to generate a new proposed sample $x'$. This proposal is accepted as the next value $x_{t+1} = x'$ if $\alpha$ drawn from $U(0, 1)$ satisfies

$$\alpha < \frac{P(x')}{P(x_t)}$$

If the proposal is not accepted, then the current value of $x$ is retained ($x_{t+1} = x_t$).

The Markov chain is started from a random initial value $x_0$ and the algorithm is run for many iterations until this initial state is forgotten. These samples, which are discarded, are known as burn-in. The remaining set of accepted values of $x$ represent a sample from the distribution $P(x)$. As a Gaussian proposal density (or a lognormal one for definite-positive parameters) is used, the variance parameter $\sigma^2_x$ has to be tuned during the burn-in period. This is done by calculating the acceptance rate, which is the fraction of proposed samples that is accepted in a window of the last $N$ samples. The desired acceptance rate depends on the target distribution, however it has been shown theoretically that the ideal acceptance rate for a one dimensional Gaussian distribution is approx 50%, decreasing to approx 23% for an N-dimensional Gaussian target distribution. If $\sigma^2_x$ is too small the chain will mix slowly (i.e., the acceptance rate will be too high, so the sampling will move around the space slowly and converge slowly to $P(x)$). If $\sigma^2_x$ is too large the acceptance rate will be very low because the proposals are likely to land in regions of much lower probability density. The desired acceptance rate is fixed here to 34%.

The MCMC algorithm is based on a code developed by Eric Gaume on Scilab. It is still unstable and not all the distributions have been tested.

**Value**

$\texttt{BayesianMCMC}$ and $\texttt{BayesianMCMCcont}$ (which just continues the simulations of $\texttt{BayesianMCMC}$ for local analyses and $\texttt{BayesianMCMCreg}$ and $\texttt{BayesianMCMCregcont}$ for regional analyses) return the following values:

- $\texttt{parameters}$ matrix (nbpas)x(nbchaines) with the simulated sets of parameters with the MCMC algorithm;
- $\texttt{parametersML}$ set of parameters correspondent to the maximum likelihood;
- $\texttt{returnperiods}$ return periods for which $\texttt{quantilesML}$ and $\texttt{intervals}$ are calculated;
- $\texttt{quantilesML}$ quantiles correspondent to $\texttt{returnperiods}$ for the distribution whose parameters are $\texttt{parametersML}$;
- $\texttt{logML}$ maximum log-likelihood;
- $\texttt{intervals}$ confidence intervals for the quantiles $\texttt{quantilesML}$ for limits $\texttt{confint}$;
- $\texttt{varparameters}$ matrix (nbpas)x(nbchaines)x(number of parameters) with the simulated variances for the MCMC algorithm;
- $\texttt{vraisdist}$ likelihoods for the sets $\texttt{parameters}$;
- $\texttt{propsaut}$ vector showing the evolution of the acceptance rate during the Bayesian MCMC fitting;
- $\texttt{plot.BayesianMCMC}$ and $\texttt{plot.BayesianMCMCreg}$ (for a normalized surface of 1 km2) plot the following figures:
1 data as plotting position (the Cunanne plotting position $a = 0.4$ is used), fitted distribution (maximum likelihood) and confidence intervals;
2 diagnostic plot of the MCMC simulation (parameters);
3 diagnostic plot of the MCMC simulation (likelihood and MCMC acceptance rate);
4 posterior distribution of parameters obtained with the MCMC simulation (cloud plots);
5 a-priori distribution of parameters (contour plots);
plotBayesianMCMCreg_surf plots the same plot as the first one given by plot.BayesianMCMCreg but for each surface in argument, as well as its mean as a function of the surfaces;

Note
For information on the package and the Author, and for all the references, see nsRFA.

Author(s)
Eric Gaume, Alberto Viglione, Jose Luis Salinas, Olivier Payrastre, Chi Cong N’guyen, Karine Halbert

See Also
.

Examples
set.seed(2988)
série <- rand.GEV(120, xi=40, alfa=20, k=-0.4)
série100 <- série[1:100]
série100[série100 < 250] <- NA
série20 <- série[101:120]
série <- c(série100, série20)

plot(série, type="h", ylim=c(0, 600), xlab="", ylab="Annual flood peaks [m3/s]", lwd=3)
abline(h=0)
points(série100, col=2)

## Not run:
# Using only systemtic data
only_sist <- BayesianMCMC (xcont=série20, nbpas=5000, nbchaines=3, varparameters0=c(70, 20, 0.5), confint=c(0.05, 0.95), dist="GEV")
plot(only_sist, which=c(1:3), ask=TRUE, ylim=c(1,600))
only_sist <- BayesianMCMCcont(only_sist)
plot(only_sist, which=c(1:3), ask=TRUE, ylim=c(1,600))
only_sist <- BayesianMCMCcont(only_sist)
plot(only_sist, which=c(1:3), ask=TRUE, ylim=c(1,600))
# Adding the information that the threshold 250 m³/s was exceeded
# 3 times in the past 100 years
with_hist_thresh <- BayesianMCMC (xcont=serie20, infhist=rep(250,3), nbans=100, seuil=250, nbpaspas=5000, nbchaines=3, confint=c(0.05, 0.95), dist="GEV")
plot(with_hist_thresh, which=c(1:3), ask=TRUE, ylim=c(1,600))

# Assuming that the 3 historical events are known with high uncertainty
with_hist_limits <- BayesianMCMC (xcont=serie20, infhist=c(320,320,250), suphist=c(360,400,270), nbans=100, seuil=250, nbpaspas=5000, nbchaines=3, confint=c(0.05, 0.95), dist="GEV")
plot(with_hist_limits, which=c(1:3), ask=TRUE, ylim=c(1,600))

# Assuming that the 3 historical events are perfectly known
with_hist_known <- BayesianMCMC (xcont=serie20, xhist=serie100[!is.na(serie100)], nbans=100, seuil=250, nbpaspas=5000, nbchaines=3, confint=c(0.05, 0.95), dist="GEV")
plot(with_hist_known, which=c(1:3), ask=TRUE, ylim=c(1,600))

# Perception threshold without available information on floods
without_info <- BayesianMCMC (xcont=serie20, xhist=-1, nbans=100, seuil=2400, nbpaspas=5000, nbchaines=3, confint=c(0.05, 0.95), dist="GEV")
plot(without_info, which=c(1:3), ask=TRUE, ylim=c(1,600))

# Using one reasonable a-priori distribution
fNORM3 <- function (x) {
  # x = vector of values
  # mu = vector of means
  mu = c(44, 26, -0.40)
  # CM = covariance matrix
  CM = matrix(c(13, 7.8, -0.055, 7.8, 15, -0.42, -0.055, -0.42, 0.056), nrow=3, ncol=3)
  CM <- solve(CM)
  term2 <- exp(-((x - mu) %*% CM %*% (x - mu))/2)
  term1 <- 1/(2*pi)^(3/2)/sqrt(det(CM))
  return(term2 * term1)
}
## BayesianMCMC

```r
term1*term2
}

with_hist_known2 <- BayesianMCMC(xcont=serie20, xhist=serie100[!is.na(serie100)],
                                   nbans=100, seuil=250,
                                   nbpas=5000, nbchaines=3, apriori=fNORM3,
                                   confint=c(0.05, 0.95), dist="GEV")

plot(with_hist_known2, 5)
plot(with_hist_known2, 4)
plot(with_hist_known, 4)
plot(with_hist_known)
plot(with_hist_known2)

# Using one non-reasonable a-priori distribution
fNORM3 <- function(x) {
  # x = vector of values
  # mu = vector of means
  mu = c(30, 50, -0.10)
  # CM = covariance matrix
  CM = matrix(c(13, 7.8, -0.055,
               7.8, 15, -0.42,
               -0.055, -0.42, 0.056), nrow=3, ncol=3)
  CMm1 <- solve(CM)
  term2 <- exp(-((x - mu) %*% CMm1 %*% (x - mu))/2)
  term2
}

with_hist_known3 <- BayesianMCMC(xcont=serie20, xhist=serie100[!is.na(serie100)],
                                   nbans=100, seuil=250,
                                   nbpas=5000, nbchaines=3, apriori=fNORM3,
                                   confint=c(0.05, 0.95), dist="GEV")

plot(with_hist_known3, 5)
plot(with_hist_known3, 4)
plot(with_hist_known, 4)
plot(with_hist_known)
plot(with_hist_known3)

## End(Not run)

## Not run:
# Assuming that the historical events are perfectly known and there are 4 different thresholds
# The data file is presenting this way:

# xhist nbans seuil
# 6000 55 6000
# 7400 28 7250
# 6350 8 3050
# 4000 0 3050
# 4550 0 3050
# 3950 0 3050
# 7550 58 2400
```
bestlm

### Description

bestlm performs an exhaustive search for the best subsets of the variables in ind for predicting dip in linear regression.
Usage

bestlm (dip, ind, kmax=4, nbest=3)
## S3 method for class 'bestlm'
print(x, ...)
## S3 method for class 'bestlm'
summary(object, ...)

Arguments

x object of class bestlm, output of function bestlm
object object of class bestlm, output of function bestlm
dip vector n x 1 of dependent variable to be predicted
ind matrix n x K of the K independent variables (candidate predictors)
kmax maximum size (number of regressors) to report
nbest number of subsets of each size to report
... other arguments

Details

This function has been obtained using the function leaps of the R package leaps. It is based on
the Alan Miller’s FORTRAN routines.
Warning: the function will stop with an error if ind is not of full rank or if it has more than 31

Value

bestlm returns the following values:
subselect matrix (kmax*nbest)x(ncol(ind)) with the sets of chosen linear models, ordered in func-
tion of the adjusted coefficient of determination (R2adj);
R2adj the ordered adjusted coefficient of determination;

Note

For information on the package and the Author, and for all the references, see nsRFA.

See Also

lm, REGRDIAGNOSTICS.

Examples

data(hydroSIMN)

bestlm(parameters[,"Dm"], parameters[-c(1:2)])

regr <- lm(Dm ~ Am + S2000 + NORD, parameters); regr
summary(regr)
Diagnostics of models

Description

Diagnostics of model results, it compares estimated values \( y \) with observed values \( x \).

Usage

\[
\begin{align*}
R^2 & = (x, y, na.rm=FALSE) \\
RMSE & = (x, y, na.rm=FALSE) \\
MAE & = (x, y, na.rm=FALSE) \\
RMSEP & = (x, y, na.rm=FALSE) \\
MAEP & = (x, y, na.rm=FALSE)
\end{align*}
\]

Arguments

- \( x \): observed values
- \( y \): estimated values
- \( na.rm \): logical. Should missing values be removed?

Details

If \( x_i \) are the observed values, \( y_i \) the estimated values, with \( i = 1, ..., n \), and \( \bar{x} \) the sample mean of \( x_i \), then:

\[
R^2 = 1 - \frac{\sum_{i=1}^{n} (x_i - y_i)^2}{\sum_{i=1}^{n} x_i^2 - n\bar{x}^2}
\]

\[
RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - y_i)^2}
\]

\[
MAE = \frac{1}{n} \sum_{i=1}^{n} |x_i - y_i|
\]

\[
RMSEP = \sqrt{\frac{1}{n} \sum_{i=1}^{n} ((x_i - y_i)/x_i)^2}
\]

\[
MAEP = \frac{1}{n} \sum_{i=1}^{n} |(x_i - y_i)/x_i|
\]

Value

\( \text{R}^2 \) returns the coefficient of determination \( R^2 \) of a model.

RMSE returns the root mean squared error of a model.

MAE returns the mean absolute error of a model.

RMSEP returns the percentual root mean squared error of a model.

MAE returns the percentual mean absolute error of a model.

Note

For information on the package and the Author, and for all the references, see \texttt{nsRFA}.

See Also

\texttt{lm, summary.lm, predict.lm, REGRDIGNOSTICS}

Examples

data(hydroSIMN)

datregr <- parameters
regr0 <- \texttt{lm(Dm ~ ., datregr)}; \texttt{summary(regr0)}
regr1 <- \texttt{lm(Dm ~ Am + Hm + Ybar, datregr)}; \texttt{summary(regr1)}

obs <- parameters[,"Dm"]
est0 <- regr0$fitted.values
est1 <- regr1$fitted.values

R2(obs, est0)
R2(obs, est1)

RMSE(obs, est0)
RMSE(obs, est1)

MAE(obs, est0)
MAE(obs, est1)

RMSEP(obs, est0)
RMSEP(obs, est1)

MAEP(obs, est0)
MAEP(obs, est1)
Description

Sample values are plotted against their empirical distribution in graphs where points belonging to a particular distribution should lie on a straight line.

Usage

plotpos (x, a=0, orient="xF", ...)  
plotposRP (x, a=0, orient="xF", ...)  
loglogplot (x, a=0, orient="xF", ...)  
unifplot (x, a=0, orient="xF", line=FALSE, ...)  
normplot (x, a=0, orient="xF", line=FALSE, ...)  
lognormplot (x, a=0, orient="xF", line=FALSE, ...)  
studentplot (x, df, a=0, orient="xF", line=FALSE, ...)  
logisplot (x, a=0, orient="xF", line=FALSE, ...)  
gammaplot (x, shape, a=0, orient="xF", line=FALSE, ...)  
expplot (x, a=0, orient="xF", line=FALSE, ...)  
paretoplot (x, a=0, orient="xF", line=FALSE, ...)  
gumbelplot (x, a=0, orient="xF", line=FALSE, ...)  
frechetplot (x, a=0, orient="xF", line=FALSE, ...)  
weibullplot (x, a=0, orient="xF", line=FALSE, ...)  
plotposRPhist (xcont, xhist=NA, infhist=NA, suphist=NA, nbans=NA, seuil=NA, col12=c(1,1), a=0, orient="xF", ...)  
pointspos (x, a=0, orient="xF", ...)  
pointsposRP (x, a=0, orient="xF", ...)  
loglogpoints (x, a=0, orient="xF", ...)  
unifpoints (x, a=0, orient="xF", ...)  
normpoints (x, a=0, orient="xF", ...)  
studentpoints (x, df, a=0, orient="xF", ...)  
logispoints (x, a=0, orient="xF", ...)  
gammapoints (x, shape, a=0, orient="xF", ...)  
exppoints (x, a=0, orient="xF", ...)  
gumbelpoints (x, a=0, orient="xF", ...)  
weibullpoints (x, a=0, orient="xF", ...)  
regionalplotpos (x, cod, a=0, orient="xF", ...)  
regionalnormplot (x, cod, a=0, orient="xF", ...)  
regionallognormplot (x, cod, a=0, orient="xF", ...)  
regionalexpplot (x, cod, a=0, orient="xF", ...)  
regionalparetoplot (x, cod, a=0, orient="xF", ...)  
regionalgumbelplot (x, cod, a=0, orient="xF", ...)  
regionalfrechetplot (x, cod, a=0, orient="xF", ...)  
pointsposRPhist (xcont, xhist=NA, infhist=NA, suphist=NA, nbans=NA, seuil=NA, col12=c(1,1), a=0, orient="xF", ...)
Arguments

- **x**: vector representing a data-sample
- **xcont**: vector of systematic data (see BayesianMCMC)
- **xhist**: vector of historical data (see BayesianMCMC)
- **infhist**: inferior limit for historical data (see BayesianMCMC)
- **suphist**: superior limit for historical data (see BayesianMCMC)
- **nbans**: period (in years) over which the threshold has not been exceeded except for the historical data (see BayesianMCMC)
- **seuil**: threshold non exceeded in the historical period except for the historical data (see BayesianMCMC)
- **df**: degrees of freedom (> 0, maybe non-integer) of the Student t distribution. 'df = Inf' is allowed.
- **shape**: shape parameter of the distribution
- **a**: plotting position parameter, normally between 0 and 0.5 (the default value here, corresponding to the Hazen plotting position, see details)
- **orient**: if orient="xF" the abscissa will be x and the ordinate F
- **line**: if TRUE (default) a straight line indicating the normal, lognormal, ..., distribution with parameters estimated from x is plotted
- **cod**: array that defines the data subdivision among sites
- **col12**: vector of 2 elements containing the colors for the systematic and historical data respectively
- **...**: graphical parameters as xlab, ylab, main, ...

Details


For the quantiles of the comparison distribution typically the Weibull formula \( k/(n + 1) \) is used (default here). Several different formulas have been used or proposed as symmetrical plotting positions. Such formulas have the form

\[
(k - a)/(n + 1 - 2a)
\]

for some value of \( a \) in the range from 0 to 1/2. The above expression \( k/(n + 1) \) is one example of these, for \( a = 0 \). The Filliben plotting position has \( a = 0.3175 \) and the Cunanne plotting position has \( a = 0.4 \) should be nearly quantile-unbiased for a range of distributions. The Hazen plotting position, widely used by engineers, has \( a = 0.5 \). The Blom’s plotting position, \( a = 3/8 \), gives nearly unbiased quantiles for the normal distribution, while the Gringeton plotting position, \( a = 0.44 \), is optimized for the largest observations from a Gumbel distribution. For the generalized Pareto, the GEV and related distributions of the Type I (Gumbel) and Weibull, \( a = 0.35 \) is suggested.

For large sample size, \( n \), there is little difference between these various expressions.
Value

Representation of the values of \( x \) vs their empirical probability function \( F \) in a cartesian, uniform, normal, lognormal or Gumbel plot. `plotpos` and `unifplot` are analogous except for the axis notation. `unifplot` has the same notation as `normplot, lognormplot, ... plotposRPhist` is analogous to `plotpos` but the frequencies \( F \) are expressed as Return Periods \( T = 1/(1 - F) \). With the default settings, \( F \) is defined with the Weibull plotting position \( F = k/(n + 1) \). The straight line (if `line=TRUE`) indicate the uniform, normal, lognormal or Gumbel distribution with parameters estimated from \( x \). The regional plots draw samples of a region on the same plot.

`pointspos`, `normpoints`, ... are the analogous of points, they can be used to add points or lines to `plotpos, normplot, ... normpoints` can be used either on `normplot or lognormplot`. `exppoints` can be used either on `expplot or paretoplot` (since the log-transformed Pareto random variable is exponentially distributed). `gumbelpoints` can be used either on `gumbelplot or frechetplot` (since the log-transformed Frechet random variable is distributed as a Gumbel).

`loglogplot` plots the logarithm of sample vs the logarithm of the empirical exceedance probability. For the log-log plot, the tail probability is represented by a straight line for power-law distributions (e.g. log-pearson, log-logistic, Frechet, ..., HEAVY TAIL), but not for the other subexponential or exponential distributions (e.g. gumbel, gamma, Pearson type III, ..., MODERATE TAIL); see El Adlouni et al. (2008).

`plotposRPhist` is based on the method in Stedinger et al. (1993, pp. 18.41-42).

Note

For information on the package and the Author, and for all the references, see `nsRFA`.

See Also

These functions are analogous to `qqnorm`; for the distributions, see `Normal, Lognormal, LOGNORM, GUMBEL`.

Examples

```r
x <- rnorm(30,10,2)
plotpos(x)
normplot(x)
normplot(x,xlab=expression(D[m]),ylab=expression(hat(F)),
   main="Normal plot",cex.main=1,font.main=1)
normplot(x,line=FALSE)

x <- rlnorm(30,log(100),log(10))
normplot(x)
lognormplot(x)

x <- rand.gumb(30,1000,100)
normplot(x)
gumbelplot(x)

x <- rnorm(30,10,2)
y <- rnorm(50,10,3)
z <- c(x,y)
```
Two parameter exponential distribution and L-moments

Description

EXP provides the link between L-moments of a sample and the two parameter exponential distribution.
Usage

f.exp (x, xi, alfa)
F.exp (x, xi, alfa)
invF.exp (F, xi, alfa)
Lmom.exp (xi, alfa)
par.exp (lambda1, lambda2)
rand.exp (numerosita, xi, alfa)

Arguments

x       vector of quantiles
xi      vector of exp location parameters
alfa    vector of exp scale parameters
F       vector of probabilities
lambda1 vector of sample means
lambda2 vector of L-variances
numerosita numeric value indicating the length of the vector to be generated

Details


**Definition**

Parameters (2): $\xi$ (lower endpoint of the distribution), $\alpha$ (scale).

Range of $x$: $\xi \leq x < \infty$.

Probability density function:

$$f(x) = \alpha^{-1} \exp\{-\frac{(x - \xi)}{\alpha}\}$$

Cumulative distribution function:

$$F(x) = 1 - \exp\{-\frac{(x - \xi)}{\alpha}\}$$

Quantile function:

$$x(F) = \xi - \alpha \log(1 - F)$$

**L-moments**

$$\lambda_1 = \xi + \alpha$$
$$\lambda_2 = \frac{1}{2} \cdot \alpha$$
$$\tau_3 = \frac{1}{3}$$
$$\tau_4 = \frac{1}{6}$$

Parameters
If $\xi$ is known, $\alpha$ is given by $\alpha = \lambda_1 - \xi$ and the L-moment, moment, and maximum-likelihood estimators are identical. If $\xi$ is unknown, the parameters are given by

$$\alpha = 2 \lambda_2$$

$$\xi = \lambda_1 - \alpha$$

For estimation based on a single sample these estimates are inefficient, but in regional frequency analysis they can give reasonable estimates of upper-tail quantiles.

Lmom.exp and par.exp accept input as vectors of equal length. In f.exp, F.exp, invF.exp and rand.exp parameters ($x_i$, alfa) must be atomic.

Value

f.exp gives the density $f$, F.exp gives the distribution function $F$, invF.exp gives the quantile function $x$, Lmom.exp gives the L-moments ($\lambda_1$, $\lambda_2$, $\tau_3$, $\tau_4$), par.exp gives the parameters ($x_i$, alfa), and rand.exp generates random deviates.

Note

For information on the package and the Author, and for all the references, see nsRFA.

See Also

rnorm, runif, GENLOGIS, GENPAR, GEV, GUMBEL, KAPPA, LOGNORM, P3; DISTPLOTS, GOFmontecarlo, Lmoments.

Examples

data(hydroSIMN)
annualflows
summary(annualflows)
x <- annualflows["dato"]
fac <- factor(annualflows["cod"])
split(x, fac)
camp <- split(x, fac)$"45"
ll <- Lmoments(camp)
parameters <- par.exp(ll[1], ll[2])
f.exp(1800, parameters$x_i, parameters$alfa)
F.exp(1800, parameters$x_i, parameters$alfa)
invF.exp(0.7870856, parameters$x_i, parameters$alfa)
Lmom.exp(parameters$x_i, parameters$alfa)
rand.exp(100, parameters$x_i, parameters$alfa)

Rll <- regionalLmoments(x, fac); Rll
parameters <- par.exp(Rll[1], Rll[2])
Lmom.exp(parameters$x_i, parameters$alfa)
FEH1000

Data-sample

Description

Flood Estimation Handbook flood peak data CD-ROM.

Usage

data(FEH1000)

Format

Data.frames:

am is the data.frame of the annual maximum flows with 3 columns: number, the code of the station; date, date of the annual maximum; year, year of the annual maximum (we consider hydrologic year: 1 October - 30 September); am, the value of the annual maximum flow [m^3/s].

cd is the data.frame of parameters of 1000 catchments with 24 columns: number, the code of the station; nominal_area, catchment drainage area [km^2]; nominal_ngr_x, basin outflow coordinates [m]; nominal_ngr_y, basin outflow coordinates [m]; ihdtm_ngr_x, basin outflow coordinates by Institute of Hydrology digital terrain model [m]; ihdtm_ngr_y, basin outflow coordinates by Institute of Hydrology digital terrain model [m]; dtm_area, catchment drainage area [km^2] derived by CEH using their DTM (IHDTM); bfihost, baseflow index derived from HOST soils data; sprhost, standard percentage runoff derived from HOST soils data; farl, index of flood attenuation due to reservoirs and lakes; saar, standard average annual rainfall 1961-1990 [mm]; rmed_1d, median annual maximum 1-day rainfall [mm]; rmed_2d, median annual maximum 2-days rainfall [mm]; rmed_1h, median annual maximum 1-hour rainfall [mm]; smdbar, mean SMD (soil moisture deficit) for the period 1961-1990 calculated from MORECS month-end values [mm]; propwet, proportion of time when soil moisture deficit <=6 mm during 1961-90, defined using MORECS; 1dp, longest drainage path [km], defined by recording the greatest distance from a catchment node to the defined outlet; dplbar, mean drainage path length [km]; altbar, mean catchment altitude [m]; dpsbar, mean catchment slope [m/km]; aspbar, index representing the dominant aspect of catchment slopes (its values increase clockwise from zero to 360, starting from the north). Mean direction of all inter-nodal slopes with north being zero; aspvar, index describing the invariability in aspect of catchment slopes. Values close to one when all slopes face a similar direction; urbext1990, extent of urban and suburban land cover in 1990 [fraction].

Note

For information on the package and the Author, and for all the references, see nsRFA.

Source

http://www.environment-agency.gov.uk/hiflowsuk/
Examples

```r
data(FEH1000)
names(cd)
am[1:20,]
```

---

**Description**

Functions for inversion calculation.

**Usage**

```r
data(functionsLaio)
```

**Format**

Data.frames:

- `df.kgev` is a data.frame with the skewness coefficient (first column) and the corresponding shape parameter of the GEV (second column)
- `df.polig` represents the poligamma function.

**Note**

For information on the package and the Author, and for all the references, see `nsRFA`.

---

**GENLOGIS**

*Three parameter generalized logistic distribution and L-moments*

**Description**

GENLOGIS provides the link between L-moments of a sample and the three parameter generalized logistic distribution.

**Usage**

```r
f.genlogis (x, xi, alfa, k)
F.genlogis (x, xi, alfa, k)
invF.genlogis (F, xi, alfa, k)
Lmom.genlogis (xi, alfa, k)
par.genlogis (lambda1, lambda2, tau3)
rand.genlogis (numerisita, xi, alfa, k)
```
Arguments

- **x**: vector of quantiles
- **xi**: vector of genlogis location parameters
- **alfa**: vector of genlogis scale parameters
- **k**: vector of genlogis shape parameters
- **F**: vector of probabilities
- **lambda1**: vector of sample means
- **lambda2**: vector of L-variances
- **tau3**: vector of L-CA (or L-skewness)
- **numerosita**: numeric value indicating the length of the vector to be generated

Details


**Definition**

Parameters (3): \( \xi \) (location), \( \alpha \) (scale), \( k \) (shape).

Range of \( x \): \(-\infty < x \leq \xi + \alpha/k \) if \( k > 0 \); \(-\infty < x < \infty \) if \( k = 0 \); \( \xi + \alpha/k \leq x < \infty \) if \( k < 0 \).

Probability density function:

\[
  f(x) = \frac{\alpha^{-1}e^{-(1-k)y}}{(1+e^{-y})^2}
\]

where \( y = -k^{-1}\log(1-k(x-\xi)/\alpha) \) if \( k \neq 0 \), \( y = (x-\xi)/\alpha \) if \( k = 0 \).

Cumulative distribution function:

\[
  F(x) = 1/(1+e^{-y})
\]

Quantile function: \( x(F) = \xi + \alpha[1 - ((1 - F)/F)^{1/k}] \) if \( k \neq 0 \), \( x(F) = \xi - \alpha\log((1 - F)/F) \) if \( k = 0 \).

\( k = 0 \) is the logistic distribution.

**L-moments**

L-moments are defined for \(-1 < k < 1\).

\[
  \lambda_1 = \xi + \alpha[1/k - \pi/\sin(k\pi)]
\]

\[
  \lambda_2 = \alpha k \pi / \sin(k\pi)
\]

\[
  \tau_3 = -k
\]

\[
  \tau_4 = (1 + 5k^2)/6
\]

**Parameters**

\( k = -\tau_3, \alpha = \frac{\lambda_2 \sin(k\pi)}{k\pi}, \xi = \lambda_1 - \alpha(\frac{1}{k} - \frac{\pi}{\sin(k\pi)}) \).

Lmom.genlogis and par.genlogis accept input as vectors of equal length. In f.genlogis, F.genlogis, invF.genlogis and rand.genlogis parameters (xi, alfa, k) must be atomic.
Value

f.genlogis gives the density \( f \), \( F \).\text{genlogis} gives the distribution function \( F \), \text{invF}.\text{genlogis} gives the quantile function \( x \). \text{Lmom}.\text{genlogis} gives the L-moments \( \lambda_1, \lambda_2, \tau_3, \tau_4 \), \text{par}.\text{genlogis} gives the parameters \( (xi, alfa, k) \), and \text{rand}.\text{genlogis} generates random deviates.

Note

For information on the package and the Author, and for all the references, see \text{nsRFA}.

See Also

\text{rnorm}, \text{runif}, \text{EXP}, \text{GENPAR}, \text{GEV}, \text{GUMBEL}, \text{KAPPA}, \text{LOGNORM}, \text{P3}; \text{DISTPLOTS}, \text{GOFmontecarlo}, \text{Lmoments}.

Examples

data(hydroSIMN)
annualflows
summary(annualflows)
x <- annualflows["dato"][,]
fac <- factor(annualflows["cod"][,])
split(x,fac)

camp <- split(x,fac)$"45"
ll <- Lmoments(camp)
parameters <- par.genlogis(ll[1],ll[2],ll[4])
f.genlogis(1800,parameters$xi,parameters$alfa,parameters$k)
F.genlogis(1800,parameters$xi,parameters$alfa,parameters$k)
invF.genlogis(0.7697433,parameters$xi,parameters$alfa,parameters$k)
Lmom.genlogis(parameters$xi,parameters$alfa,parameters$k)
rand.genlogis(100,parameters$xi,parameters$alfa,parameters$k)

Rll <- regionalLmoments(x,fac); Rll
parameters <- par.genlogis(Rll[1],Rll[2],Rll[4])
Lmom.genlogis(parameters$xi,parameters$alfa,parameters$k)

GENPAR

Three parameter generalized Pareto distribution and L-moments

Description

GENPAR provides the link between L-moments of a sample and the three parameter generalized Pareto distribution.

Usage

f.genpar (x, xi, alfa, k)
F.genpar (x, xi, alfa, k)
invF.genpar (F, xi, alfa, k)
Lmom.genpar (xi, alfa, k)
par.genpar (lambda1, lambda2, tau3)
rand.genpar (numerosita, xi, alfa, k)

Arguments

- x: vector of quantiles
- xi: vector of genpar location parameters
- alfa: vector of genpar scale parameters
- k: vector of genpar shape parameters
- F: vector of probabilities
- lambda1: vector of sample means
- lambda2: vector of L-variances
- tau3: vector of L-CA (or L-skewness)
- numerosita: numeric value indicating the length of the vector to be generated

Details


**Definition**

Parameters (3): $\xi$ (location), $\alpha$ (scale), $k$ (shape).

Range of $x$: $\xi < x \leq \xi + \alpha/k$ if $k > 0$; $\xi \leq x < \infty$ if $k \leq 0$.

Probability density function:

$$f(x) = \alpha^{-1} e^{-(1-k)y}$$

where $y = -k^{-1} \log\{1 - k(x - \xi)/\alpha\}$ if $k \neq 0$, $y = (x - \xi)/\alpha$ if $k = 0$.

Cumulative distribution function:

$$F(x) = 1 - e^{-y}$$

Quantile function: $x(F) = \xi + \alpha[1 - (1 - F)^k]/k$ if $k \neq 0$, $x(F) = \xi - \alpha \log(1 - F)$ if $k = 0$.

$k = 0$ is the exponential distribution; $k = 1$ is the uniform distribution on the interval $\xi < x \leq \xi + \alpha$.

**L-moments**

L-moments are defined for $k > -1$.

$$\lambda_1 = \xi + \alpha/(1 + k)$$
$$\lambda_2 = \alpha/[(1 + k)(2 + k)]$$
$$\tau_3 = (1 - k)/(3 + k)$$
$$\tau_4 = (1 - k)(2 - k)/[(3 + k)(4 + k)]$$

The relation between $\tau_3$ and $\tau_4$ is given by

$$\tau_4 = \frac{\tau_3(1 + 5\tau_3)}{5 + \tau_3}$$
Parameters

If $\xi$ is known, $k = (\lambda_1 - \xi)/\lambda_2 - 2$ and $\alpha = (1+k)(\lambda_1 - \xi)$; if $\xi$ is unknown, $k = (1-3\tau_3)/(1+\tau_3)$, $\alpha = (1+k)(2+k)\lambda_2$ and $\xi = \lambda_1 - (2+k)\lambda_2$.

Lmom.genpar and par.genpar accept input as vectors of equal length. In f.genpar, F.genpar, invF.genpar and rand.genpar parameters ($x_i$, alfa, $k$) must be atomic.

Value

f.genpar gives the density $f$, F.genpar gives the distribution function $F$, invF.genpar gives the quantile function $x$. Lmom.genpar gives the L-moments ($\lambda_1$, $\lambda_2$, $\tau_3$, $\tau_4$), par.genpar gives the parameters ($x_i$, alfa, $k$), and rand.genpar generates random deviates.

Note

For information on the package and the Author, and for all the references, see nsRFA.

See Also

rnorm, runif, EXP, GENLOGIS, GEV, GUMBEL, KAPPA, LOGNORM, P3; DISTPLOTS, GOFmontecarlo, Lmoments.

Examples

data(hydroSIMN)
annualflows
summary(annualflows)
x <- annualflows["dato"][,] 
fac <- factor(annualflows["cod"][,] 
split(x,fac)

x <- split(x,fac)$"45"
ll <- Lmoments(camp)
parameters <- par.genpar(ll[1],ll[2],ll[4])
f.genpar(1800,parameters$x_i,parameters$alfa,parameters$k)
F.genpar(1800,parameters$x_i,parameters$alfa,parameters$k)
invF.genpar(0.7161775,parameters$x_i,parameters$alfa,parameters$k)
Lmom.genpar(parameters$x_i,parameters$alfa,parameters$k)
rand.genpar(100,parameters$x_i,parameters$alfa,parameters$k)

Rll <- regionalLmoments(x,fac); Rll
parameters <- par.genpar(Rll[1],Rll[2],Rll[4])
Lmom.genpar(parameters$x_i,parameters$alfa,parameters$k)
### GEV

Three parameter generalized extreme value distribution and L-moments

#### Description

GEV provides the link between L-moments of a sample and the three parameter generalized extreme value distribution.

#### Usage

- `f.GEV(x, xi, alfa, k)`
- `F.GEV(x, xi, alfa, k)`
- `invF.GEV(F, xi, alfa, k)`
- `Lmom.GEV(xi, alfa, k)`
- `par.GEV(lambda1, lambda2, tau3)`
- `rand.GEV(numerosita, xi, alfa, k)`

#### Arguments

- `x` vector of quantiles
- `xi` vector of GEV location parameters
- `alfa` vector of GEV scale parameters
- `k` vector of GEV shape parameters
- `F` vector of probabilities
- `lambda1` vector of sample means
- `lambda2` vector of L-variances
- `tau3` vector of L-CA (or L-skewness)
- `numerosita` numeric value indicating the length of the vector to be generated

#### Details


#### Definition

Parameters (3): ξ (location), α (scale), k (shape).

**Range of x:** $-\infty < x \leq \xi + \alpha/k$ if $k > 0$; $-\infty < x < \infty$ if $k = 0$; $\xi + \alpha/k \leq x < \infty$ if $k < 0$.

**Probability density function:**

$$f(x) = \alpha^{-1}e^{-(1-k)y}e^{-y}$$

where $y = -k^{-1}\log\{1 - k(x - \xi)/\alpha\}$ if $k \neq 0$, $y = (x - \xi)/\alpha$ if $k = 0$.

**Cumulative distribution function:**

$$F(x) = e^{-e^{-y}}$$
Quantile function: \( x(F) = \xi + \alpha [1 - (\log F)^k]/k \) if \( k \neq 0 \), \( x(F) = \xi - \alpha \log(-\log F) \) if \( k = 0 \). 
\( k = 0 \) is the Gumbel distribution; \( k = 1 \) is the reverse exponential distribution.

**L-moments**
L-moments are defined for \( k > -1 \).

\[
\lambda_1 = \xi + \alpha [1 - \Gamma(1 + k)]/k \\
\lambda_2 = \alpha (1 - 2^{-k}) \Gamma(1 + k)/k \\
\tau_3 = 2(1 - 3^{-k})/(1 - 2^{-k}) - 3 \\
\tau_4 = [5(1 - 4^{-k}) - 10(1 - 3^{-k}) + 6(1 - 2^{-k})]/(1 - 2^{-k})
\]

Here \( \Gamma \) denote the gamma function
\[
\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt
\]

**Parameters**
To estimate \( k \), no explicit solution is possible, but the following approximation has accuracy better than \( 9 \times 10^{-4} \) for \(-0.5 \leq \tau_3 \leq 0.5\):

\[
k \approx 7.8590c + 2.9554c^2
\]

where
\[
c = \frac{2}{3 + \tau_3} \frac{\log 2}{\log 3}
\]

The other parameters are then given by

\[
\alpha = \frac{\lambda_2 k}{(1 - 2^{-k}) \Gamma(1 + k)} \\
\xi = \lambda_1 - \alpha [1 - \Gamma(1 + k)]/k
\]

Lmom.GEV and par.GEV accept input as vectors of equal length. In f.GEV, F.GEV, invF.GEV and rand.GEV parameters \((xi, alfa, k)\) must be atomic.

**Value**

f.GEV gives the density \( f \), F.GEV gives the distribution function \( F \), invF.GEV gives the quantile function \( x \), Lmom.GEV gives the L-moments \((\lambda_1, \lambda_2, \tau_3, \tau_4)\), par.GEV gives the parameters \((xi, alfa, k)\), and rand.GEV generates random deviates.

**Note**

For information on the package and the Author, and for all the references, see nsRFA.

**See Also**

rnorm, runif, EXP, GENLOGIS, GENPAR, GUMBEL, KAPPA, LOGNORM, P3; DISTPLOTS, GOFmontecarlo, Lmoments.
Examples

data(hydroSIMN)
annualflows
summary(annualflows)
x <- annualflows["dato"][,]
fac <- factor(annualflows["cod"][,])
split(x,fac)

camp <- split(x,fac)$"45"
ll <- Lmoments(camp)
parameters <- par.GEV(ll[1],ll[2],ll[4])
f.GEV(1800,parameters$xi,parameters$alfa,parameters$k)
F.GEV(1800,parameters$xi,parameters$alfa,parameters$k)
invF.GEV(0.7518357,parameters$xi,parameters$alfa,parameters$k)
Lmom.GEV(parameters$xi,parameters$alfa,parameters$k)
rand.GEV(100,parameters$xi,parameters$alfa,parameters$k)

Rll <- regionalLmoments(x,fac); Rll
parameters <- par.GEV(Rll[1],Rll[2],Rll[4])
Lmom.GEV(parameters$xi,parameters$alfa,parameters$k)

GOFlaio2004

Goodness of fit tests

Description

Anderson-Darling goodness of fit tests for extreme-value distributions, from Laio (2004).

Usage

A2_GOFlaio (x, dist="NORM")
A2 (F)
W2 (F)
fw2 (w)

Arguments

x data sample
F cumulative distribution function (that has to be sorted increasingly)
w Transformed test statistic (Laio, 2004)
Details


Value

A2_GOFlaio tests the goodness of fit of a distribution with the sample x; it return the value $A_2$ of the Anderson-Darling statistics and its non-exceedence probability $P(A_2)$. Note that $P$ is the probability of obtaining the test statistic $A_2$ lower than the one that was actually observed, assuming that the null hypothesis is true, i.e., $P$ is one minus the p-value usually employed in statistical testing (see http://en.wikipedia.org/wiki/P-value). If $P(A_2)$ is, for example, greater than 0.90, the null hypothesis at significance level $\alpha = 10\%$ is rejected.

A2 is the Anderson-Darling test statistic; it is used by A2_GOFlaio.

W2 is the Cramer-von Mises test statistic.

fw2 is the approximation of the probability distribution of w (first 2 terms) when $H_0$ is true (Anderson-Darling, 1952); it is used by A2_GOFlaio.

Note

For information on the package and the Author, and for all the references, see nsRFA.

See Also

GOFmontecarlo, MLlaio2004.

Examples

```r
sm <- rand.gumb(100, 0, 1)
ml <- ML_estimation (sm, dist="GEV"); ml
F.GEV(sm, ml[1], ml[2], ml[3])
A2(sort(F.GEV(sm, ml[1], ml[2], ml[3])))
A2_GOFlaio(sm, dist="GEV")

ml <- ML_estimation (sm, dist="P3"); ml
A2(sort(sort(F.gamma(sm, ml[1], ml[2], ml[3]))))
A2_GOFlaio(sm, dist="P3")
```

---

**GOFmontecarlo**

Goodness of fit tests

Description

Anderson-Darling goodness of fit tests for Regional Frequency Analysis: Monte-Carlo method.
Usage

- `gofNORMtest (x)`
- `gofEXPtest (x, Nsim=1000)`
- `gofGUMBELtest (x, Nsim=1000)`
- `gofGENLOGISTest (x, Nsim=1000)`
- `gofGENPARTest (x, Nsim=1000)`
- `gofGEVtest (x, Nsim=1000)`
- `gofLOGNORMtest (x, Nsim=1000)`
- `gofP3test (x, Nsim=1000)`

Arguments

- `x` data sample
- `Nsim` number of simulated samples from the hypothetical parent distribution

Details

An introduction, analogous to the following one, on the Anderson-Darling test is available on [http://en.wikipedia.org/wiki/Anderson-Darling_test](http://en.wikipedia.org/wiki/Anderson-Darling_test).

Given a sample \( x_i \) (\( i = 1, \ldots, m \)) of data extracted from a distribution \( F_R(x) \), the test is used to check the null hypothesis \( H_0 : F_R(x) = F(x, \theta) \), where \( F(x, \theta) \) is the hypothetical distribution and \( \theta \) is an array of parameters estimated from the sample \( x_i \).

The Anderson-Darling goodness of fit test measures the departure between the hypothetical distribution \( F(x, \theta) \) and the cumulative frequency function \( F_m(x) \) defined as:

\[
F_m(x) = \begin{cases} 
0, & x < x_{(1)} \\
\frac{i}{m}, & x_{(i)} \leq x < x_{(i+1)} \\
1, & x_{(m)} \leq x 
\end{cases}
\]

where \( x_{(i)} \) is the \( i \)-th element of the ordered sample (in increasing order).

The test statistic is:

\[
Q^2 = m \int_x [F_m(x) - F(x, \theta)]^2 \Psi(x) \, dF(x)
\]

where \( \Psi(x) \), in the case of the Anderson-Darling test (Laio, 2004), is \( \Psi(x) = [F(x, \theta)(1 - F(x, \theta))]^{-1} \). In practice, the statistic is calculated as:

\[
A^2 = -m - \frac{1}{m} \sum_{i=1}^{m} \left\{ (2i - 1) \ln[F(x_{(i)}, \theta)] + (2m + 1 - 2i) \ln[1 - F(x_{(i)}, \theta)] \right\}
\]

The statistic \( A^2 \), obtained in this way, may be confronted with the population of the \( A^2 \)'s that one obtain if samples effectively belongs to the \( F(x, \theta) \) hypothetical distribution. In the case of the test of normality, this distribution is defined (see Laio, 2004). In other cases, e.g. the Pearson Type III case, can be derived with a Monte-Carlo procedure.
gofNORMtest tests the goodness of fit of a normal (Gauss) distribution with the sample x.
gofEXPtest tests the goodness of fit of an exponential distribution with the sample x.
gofGUMBELtest tests the goodness of fit of a Gumbel (EV1) distribution with the sample x.
gofGENLOGIStest tests the goodness of fit of a Generalized Logistic distribution with the sample x.
gofGENPARtest tests the goodness of fit of a Generalized Pareto distribution with the sample x.
gofGEVtest tests the goodness of fit of a Generalized Extreme Value distribution with the sample x.
gofLOGNORMtest tests the goodness of fit of a 3 parameters Lognormal distribution with the sample x.
gofP3test tests the goodness of fit of a Pearson type III (gamma) distribution with the sample x.

They return the value $A_2$ of the Anderson-Darling statistics and its non exceedence probability $P$. Note that $P$ is the probability of obtaining the test statistic $A_2$ lower than the one that was actually observed, assuming that the null hypothesis is true, i.e., $P$ is one minus the p-value usually employed in statistical testing (see [http://en.wikipedia.org/wiki/P-value](http://en.wikipedia.org/wiki/P-value)). If $P(A_2)$ is, for example, greater than 0.90, the null hypothesis at significance level $\alpha = 10\%$ is rejected.

Note

For information on the package and the Author, and for all the references, see nsRFA.

See Also

traceWminim, roi, HOMTESTS.

Examples

```r
x <- rnorm(30,10,1)
gofNORMtest(x)

x <- rand.gamma(50, 100, 15, 7)
gofP3test(x, Nsim=200)

x <- rand.GEV(50, 0.907, 0.169, 0.0304)
gofGEVtest(x, Nsim=200)

x <- rand.genlogis(50, 0.907, 0.169, 0.0304)
gofGENLOGIStest(x, Nsim=200)

x <- rand.genpar(50, 0.716, 0.418, 0.476)
gofGENPARtest(x, Nsim=200)

x <- rand.lognorm(50, 0.716, 0.418, 0.476)
gofLOGNORMtest(x, Nsim=200)
```
Two parameter Gumbel distribution and L-moments

Description
GUMBEL provides the link between L-moments of a sample and the two parameter Gumbel distribution.

Usage
- f.gumb (x, xi, alfa)
- F.gumb (x, xi, alfa)
- invF.gumb (F, xi, alfa)
- Lmom.gumb (xi, alfa)
- par.gumb (lambda1, lambda2)
- rand.gumb (numerosita, xi, alfa)

Arguments
- x: vector of quantiles
- xi: vector of gumb location parameters
- alfa: vector of gumb scale parameters
- F: vector of probabilities
- lambda1: vector of sample means
- lambda2: vector of L-variances
- numerosita: numeric value indicating the length of the vector to be generated

Details

Definition
Parameters (2): ξ (location), α (scale).
Range of x: −∞ < x < ∞.
Probability density function:

\[ f(x) = \alpha^{-1} \exp[-(x - \xi)/\alpha] \exp[- \exp[-(x - \xi)/\alpha]] \]

Cumulative distribution function:

\[ F(x) = \exp[- \exp(-(x - \xi)/\alpha)] \]

Quantile function: \[ x(F) = \xi - \alpha \log(- \log F). \]

L-moments
\[
\lambda_1 = \xi + \alpha \gamma \\
\lambda_2 = \alpha \log 2 \\
\tau_3 = 0.1699 = \log(9/8)/\log 2 \\
\tau_4 = 0.1504 = (16 \log 2 - 10 \log 3)/\log 2
\]

Here \( \gamma \) is Euler's constant, 0.5772...

**Parameters**

\[
\alpha = \lambda_2/\log 2 \\
\xi = \lambda_1 - \gamma \alpha
\]

`Lmom.gumb` and `par.gumb` accept input as vectors of equal length. In `f.gumb`, `F.gumb`, `invF.gumb` and `rand.gumb` parameters \((\xi, \alpha)\) must be atomic.

**Value**

`f.gumb` gives the density \( f \), `F.gumb` gives the distribution function \( F \), `invF.gumb` gives the quantile function \( x \), `Lmom.gumb` gives the L-moments \((\lambda_1, \lambda_2, \tau_3, \tau_4)\), `par.gumb` gives the parameters \((\xi, \alpha)\), and `rand.gumb` generates random deviates.

**Note**

For information on the package and the Author, and for all the references, see `nsRFA`.

**See Also**

`rnorm`, `runif`, `EXP`, `GENLOGIS`, `GENPAR`, `GEV`, `KAPPA`, `LOGNORM`, `P3`, `DISTPLOTS`, `GOFmontecarlo`, `Lmoments`.

**Examples**

```r
data(hydroSIMN)
annualflows[1:10,]
summary(annualflows)
x <- annualflows["dato"][,]
fac <- factor(annualflows["cod"][,])
split(x,fac)
camp <- split(x,fac)$"45"
ll <- Lmoments(camp)
parameters <- par.gumb(ll[1],ll[2])
f.gumb(1800,parameters$xi,parameters$alfa)
F.gumb(1800,parameters$xi,parameters$alfa)
invF.gumb(0.7686843,parameters$xi,parameters$alfa)
Lmom.gumb(parameters$xi,parameters$alfa)
rand.gumb(100,parameters$xi,parameters$alfa)
```
Rll <- regionalLmoments(x,fac)
parameters <- par.gumb(Rll[1],Rll[2])
Lmom.gumb(parameters$xi,parameters$alfa)

**HOMTESTS**

### Homogeneity tests

**Description**

Homogeneity tests for Regional Frequency Analysis.

**Usage**

- `ADbootstrap.test (x, cod, Nsim=500, index=2)`
- `HW.tests (x, cod, Nsim=500)`
- `DK.test (x, cod)`
- `discordancy (x, cod)`
- `criticalD ()`

**Arguments**

- `x` vector representing data from many samples defined with `cod`
- `cod` array that defines the data subdivision among sites
- `Nsim` number of regions simulated with the bootstrap of the original region
- `index` if `index=1` samples are divided by their average value; if `index=2` (default) samples are divided by their median value

**Details**

**The Hosking and Wallis heterogeneity measures**

The idea underlying Hosking and Wallis (1993) heterogeneity statistics is to measure the sample variability of the L-moment ratios and compare it to the variation that would be expected in a homogeneous region. The latter is estimated through repeated simulations of homogeneous regions with samples drawn from a four parameter kappa distribution (see e.g., Hosking and Wallis, 1997, pp. 202-204). More in detail, the steps are the following: with regards to the `k` samples belonging to the region under analysis, find the sample L-moment ratios (see, Hosking and Wallis, 1997) pertaining to the `i`-th site: these are the L-coefficient of variation (L-CV),

\[
t_i^{(1)} = \frac{1}{n_i} \sum_{j=1}^{n_i} \left( \frac{2(j-1)}{n_i-1} - 1 \right) Y_{i,j}
\]

the coefficient of L-skewness,

\[
t_i^{(3)} = \frac{1}{n_i} \sum_{j=1}^{n_i} \left( \frac{6(j-1)(j-2)}{(n_i-1)(n_i-2)} - \frac{6(j-1)}{n_i-1} + 1 \right) Y_{i,j}
\]

where \(Y_{i,j}\) is the L-moment ratio pertaining to the `i`-th sample and the `j`-th site.
and the coefficient of L-kurtosis

\[ t_4^{(i)} = \frac{1}{n_i} \sum_{j=1}^{n_i} \left( \frac{20(j-1)(j-2)(j-3)}{(n_i-1)(n_i-2)(n_i-3)} - \frac{30(j-1)(j-2)}{(n_i-1)(n_i-2)} + \frac{12(j-1)}{(n_i-1)} - 1 \right) Y_{i,j} \]

\[ \frac{1}{n_i} \sum_{j=1}^{n_i} \left( \frac{2(j-1)}{(n_i-1)} - 1 \right) Y_{i,j} \]

Note that the L-moment ratios are not affected by the normalization by the index value, i.e. it is the same to use \( X_{i,j} \) or \( Y_{i,j} \) in Equations.

Define the regional averaged L-CV, L-skewness and L-kurtosis coefficients,

\[ t^R = \frac{\sum_{i=1}^{k} n_i t^{(i)}}{\sum_{i=1}^{k} n_i} \]

\[ t^R_3 = \frac{\sum_{i=1}^{k} n_i t^{(i)}_3}{\sum_{i=1}^{k} n_i} \]

\[ t^R_4 = \frac{\sum_{i=1}^{k} n_i t^{(i)}_4}{\sum_{i=1}^{k} n_i} \]

and compute the statistic

\[ V = \left\{ \frac{\sum_{i=1}^{k} n_i (t^{(i)} - t^R)^2}{\sum_{i=1}^{k} n_i} \right\}^{1/2} \]

Fit the parameters of a four-parameters kappa distribution to the regional averaged L-moment ratios \( t^R, t^R_3 \) and \( t^R_4 \), and then generate a large number \( N_{sim} \) of realizations of sets of \( k \) samples. The \( i \)-th site sample in each set has a kappa distribution as its parent and record length equal to \( n_i \). For each simulated homogeneous set, calculate the statistic \( V \), obtaining \( N_{sim} \) values. On this vector of \( V \) values determine the mean \( \mu_V \) and standard deviation \( \sigma_V \) that relate to the hypothesis of homogeneity (actually, under the composite hypothesis of homogeneity and kappa parent distribution).

An heterogeneity measure, which is called here \( HW_1 \), is finally found as

\[ \theta_{HW_1} = \frac{V - \mu_V}{\sigma_V} \]

\( \theta_{HW_1} \) can be approximated by a normal distributed with zero mean and unit variance: following Hosking and Wallis (1997), the region under analysis can therefore be regarded as ‘acceptably homogeneous’ if \( \theta_{HW_1} < 1 \), ‘possibly heterogeneous’ if \( 1 \leq \theta_{HW_1} < 2 \), and ‘definitely heterogeneous’ if \( \theta_{HW_1} \geq 2 \). Hosking and Wallis (1997) suggest that these limits should be treated as useful guidelines. Even if the \( \theta_{HW_1} \) statistic is constructed like a significance test, significance levels obtained from such a test would in fact be accurate only under special assumptions: to have independent data both serially and between sites, and the true regional distribution being kappa.

Hosking and Wallis (1993) also give alternative heterogeneity measures (that we call \( HW_2 \) and \( HW_3 \)), in which \( V \) is replaced by:

\[ V_2 = \sum_{i=1}^{k} n_i \left\{ (t^{(i)} - t^R)^2 + (t^{(i)}_3 - t^R_3)^2 \right\}^{1/2} / \sum_{i=1}^{k} n_i \]

or

\[ V_3 = \sum_{i=1}^{k} n_i \left\{ (t^{(i)}_3 - t^R_3)^2 + (t^{(i)}_4 - t^R_4)^2 \right\}^{1/2} / \sum_{i=1}^{k} n_i \]
The test statistic in this case becomes

$$\theta_{HW_2} = \frac{V_2 - \mu V_2}{\sigma V_2}$$

or

$$\theta_{HW_3} = \frac{V_3 - \mu V_3}{\sigma V_3}$$

with similar acceptability limits as the $HW_1$ statistic. Hosking and Wallis (1997) judge $\theta_{HW_2}$ and $\theta_{HW_3}$ to be inferior to $\theta_{HW_1}$ and say that it rarely yields values larger than 2 even for grossly heterogeneous regions.

The bootstrap Anderson-Darling test

A test that does not make any assumption on the parent distribution is the Anderson-Darling ($AD$) rank test (Scholz and Stephens, 1987). The $AD$ test is the generalization of the classical Anderson-Darling goodness of fit test (e.g., D’Agostino and Stephens, 1986), and it is used to test the hypothesis that $k$ independent samples belong to the same population without specifying their common distribution function.

The test is based on the comparison between local and regional empirical distribution functions. The empirical distribution function, or sample distribution function, is defined by $F(x) = \frac{j}{\eta}, x(x) \leq x < x((j+1)$, where $\eta$ is the size of the sample and $x((j)$ are the order statistics, i.e. the observations arranged in ascending order. Denote the empirical distribution function of the $i$-th sample (local) by $\hat{F}_i(x)$, and that of the pooled sample of all $N = n_1 + \ldots + n_k$ observations (regional) by $H_N(x)$.

The $k$-sample Anderson-Darling test statistic is then defined as

$$\theta_{AD} = \sum_{i=1}^{k} n_i \int_{all x} \frac{[\hat{F}_i(x) - H_N(x)]^2}{H_N(x)[1 - H_N(x)]} dH_N(x)$$

If the pooled ordered sample is $Z_1 < \ldots < Z_N$, the computational formula to evaluate $\theta_{AD}$ is:

$$\theta_{AD} = \frac{1}{N} \sum_{i=1}^{k} \frac{1}{n_i} \sum_{j=1}^{N-1} \frac{(NM_{ij} - jn_i)^2}{j(N-j)}$$

where $M_{ij}$ is the number of observations in the $i$-th sample that are not greater than $Z_j$. The homogeneity test can be carried out by comparing the obtained $\theta_{AD}$ value to the tabulated percentage points reported by Scholz and Stephens (1987) for different significance levels.

The statistic $\theta_{AD}$ depends on the sample values only through their ranks. This guarantees that the test statistic remains unchanged when the samples undergo monotonic transformations, an important stability property not possessed by $HW$ heterogeneity measures. However, problems arise in applying this test in a common index value procedure. In fact, the index value procedure corresponds to dividing each site sample by a different value, thus modifying the ranks in the pooled sample. In particular, this has the effect of making the local empirical distribution functions much more similar to the other, providing an impression of homogeneity even when the samples are highly heterogeneous. The effect is analogous to that encountered when applying goodness-of-fit tests to distributions whose parameters are estimated from the same sample used for the test (e.g., D’Agostino and Stephens, 1986; Laio, 2004). In both cases, the percentage points for the test should be opportunely redetermined. This can be done with a nonparametric bootstrap approach presenting the following steps: build up the pooled sample $S$ of the observed non-dimensional data.
Sample with replacement from $S$ and generate $k$ artificial local samples, of size $n_1, \ldots, n_k$. Divide each sample for its index value, and calculate $\theta^{(1)}_{AD}$. Repeat the procedure for $N_{sim}$ times and obtain a sample of $\theta^{(j)}_{AD}, j = 1, \ldots, N_{sim}$ values, whose empirical distribution function can be used as an approximation of $G_{H_0}(\theta_{AD})$, the distribution of $\theta_{AD}$ under the null hypothesis of homogeneity. The acceptance limits for the test, corresponding to any significance level $\alpha$, are then easily determined as the quantiles of $G_{H_0}(\theta_{AD})$ corresponding to a probability $(1 - \alpha)$.

We will call the test obtained with the above procedure the bootstrap Anderson-Darling test, hereafter referred to as $AD$.

**Durbin and Knott test**

The last considered homogeneity test derives from a goodness-of-fit statistic originally proposed by Durbin and Knott (1971). The test is formulated to measure discrepancies in the dispersion of the samples, without accounting for the possible presence of discrepancies in the mean or skewness of the data. Under this aspect, the test is similar to the $HW_1$ test, while it is analogous to the $AD$ test for the fact that it is a rank test. The original goodness-of-fit test is very simple: suppose to have a sample $X_i, i = 1, \ldots, n$, with hypothetical distribution $F(x)$; under the null hypothesis the random variable $F(X_i)$ has a uniform distribution in the $(0, 1)$ interval, and the statistic $D = \sum_{i=1}^{n} \cos[2\pi F(X_i)]$ is approximately normally distributed with mean 0 and variance 1 (Durbin and Knott, 1971). $D$ serves the purpose of detecting discrepancy in data dispersion: if the variance of $X_i$ is greater than that of the hypothetical distribution $F(x)$, $D$ is significantly greater than 0, while $D$ is significantly below 0 in the reverse case. Differences between the mean (or the median) of $X_i$ and $F(x)$ are instead not detected by $D$, which guarantees that the normalization by the index value does not affect the test.

The extension to homogeneity testing of the Durbin and Knott ($DK$) statistic is straightforward: we substitute the empirical distribution function obtained with the pooled observed data, $H_N(x)$, for $F(x)$ in $D$, obtaining at each site a statistic

$$D_i = \sum_{j=1}^{n_i} \cos[2\pi H_N(X_j)]$$

which is normal under the hypothesis of homogeneity. The statistic $\theta_{DK} = \sum_{i=1}^{k} D_i^2$ has then a chi-squared distribution with $k - 1$ degrees of freedom, which allows one to determine the acceptability limits for the test, corresponding to any significance level $\alpha$.

**Comparison among tests**

The comparison (Viglione et al, 2007) shows that the Hosking and Wallis heterogeneity measure $HW_1$ (only based on L-CV) is preferable when skewness is low, while the bootstrap Anderson-Darling test should be used for more skewed regions. As for $HW_2$, the Hosking and Wallis heterogeneity measure based on L-CV and L-CA, it is shown once more how much it lacks power.

Our suggestion is to guide the choice of the test according to a compromise between power and Type I error of the $HW_1$ and $AD$ tests. The L-moment space is divided into two regions: if the $t_3^R$ coefficient for the region under analysis is lower than 0.23, we propose to use the Hosking and Wallis heterogeneity measure $HW_1$; if $t_3^R > 0.23$, the bootstrap Anderson-Darling test is preferable.

**Value**

`AD bootstrap.test` and `DK.test` test gives its test statistic and its distribution value $P$. If $P$ is,
for example, 0.92, samples shouldn’t be considered heterogeneous with significance level minor of 8%.

`HW.tests` gives the two Hosking and Wallis heterogeneity measures $H_1$ and $H_2$; following Hosking and Wallis (1997), the region under analysis can therefore be regarded as ‘acceptably homogeneous’ if $H_1 < 1$, ‘possibly heterogeneous’ if $1 \leq H_1 < 2$, and ‘definitely heterogeneous’ if $H \geq 2$.

discordancy returns the discordancy measure $D$ of Hosking and Wallis for all sites. Hosking and Wallis suggest to consider a site discordant if $D \geq 3$ if $N \geq 15$ (where $N$ is the number of sites considered in the region). For $N < 15$ the critical values of $D$ can be listed with `criticalD`.

**Note**

For information on the package and the Author, and for all the references, see `nsRFA`.

**See Also**

`traceWminim`, `roi`, `KAPPA`, `HW.original`.

**Examples**

```r
data(hydroSIMN)
annualflows
summary(annualflows)
x <- annualflows["dato"],
cod <- annualflows["cod"],
split(x,cod)

#ADbootstrap.test(x,cod,Nsim=100)  # it takes some time
#HW.tests(x,cod)                  # it takes some time
DK.test(x,cod)

fac <- factor(annualflows["cod"],levels=c(34:38))
x2 <- annualflows[!is.na(fac),"dato"]
cod2 <- annualflows[!is.na(fac),"cod"]

ADbootstrap.test(x2,cod2,Nsim=100)
ADbootstrap.test(x2,cod2,index=1,Nsim=200)
HW.tests(x2,cod2,Nsim=100)
DK.test(x2,cod2)

discordancy(x,cod)
criticalD()
```

**HW.original**

*Original Hosking and Wallis Fortran routine*

**Description**

The original Fortran routine by Hosking is here used to analyse a region.
Usage

HW.original (data, cod, Nsim=500)
## S3 method for class 'HWorig'
print(x, ...)
## S3 method for class 'HWorig'
plot(x, interactive=TRUE, ...)
LMR (PARA, distr="EXP")
PEL (XMOM, distr="EXP")
SAMLMR (X, A=0, B=0)
SAMLMU (X)
SAMPWM (X, A=0, B=0)
REGLMR (data, cod)
REGTST (data, cod, A=0, B=0, Nsim=500)

Arguments

x object of class HWorig
data vector representing data from many samples defined with cod
cod array that defines the data subdivision among sites
Nsim number of regions simulated with the bootstrap of the original region
interactive logical: if TRUE the graphic showing is interactive
... additional parameter for print
PARA parameters of the distribution (vector)
distr distribution: EXP = Exponential (2 parameters: xi, alfa); GAM = Gamma (2 parameters: alfa, beta); GEV = Generalized extreme value (3 parameters: xi, alfa, k); GLO = Generalized logistic (3 parameters: xi, alfa, k); GNO = Generalized Normal (3 parameters: xi, alfa, k); GPA = Generalized Pareto (3 parameters: xi, alfa, k); GUM = Gumbel (2 parameters: xi, alfa); KAP = Kappa (4 parameters: xi, alfa, k, h); NOR = Normal (2 parameters: mu, sigma); PE3 = Pearson type III (3 parameters: mu, sigma, gamm); WAK = Wakeby (5 parameters: xi, alfa, beta, gamm, delta).
XMOM the L-moment ratios of the distribution, in order λ₁, λ₂, τ₃, τ₄, τ₅...
X a data vector
A, B Parameters of plotting position: for unbiased estimates (of the λ’s) set A=B=zero. Otherwise, plotting-position estimators are used, based on the plotting position (j+a)/(n+b) for the j’th smallest of n observations. For example, A=-0.35 and B=0.0 yields the estimators recommended by Hosking et al. (1985, technology). Details

Differences among \textit{HW.original} and \textit{HW.tests} should depend on differences among \texttt{PEL} and \texttt{par.kappa} for the kappa distribution. A numerical algorithm is used to resolve the implicit Equations (A.99) and (A.100) in Hosking and Wallis (1997, pag. 203-204). The algorithms in \texttt{PEL} and \texttt{par.kappa} are different. Anyway the results of the tests should converge asymptotically.

**IBM software disclaimer**

LMOMENTS: Fortran routines for use with the method of L-moments

Permission to use, copy, modify and distribute this software for any purpose and without fee is hereby granted, provided that this copyright and permission notice appear on all copies of the software. The name of the IBM Corporation may not be used in any advertising or publicity pertaining to the use of the software. IBM makes no warranty or representations about the suitability of the software for any purpose. It is provided "AS IS" without any express or implied warranty, including the implied warranties of merchantability, fitness for a particular purpose and non-infringement. IBM shall not be liable for any direct, indirect, special or consequential damages resulting from the loss of use, data or projects, whether in an action of contract or tort, arising out of or in connection with the use or performance of this software.

**Value**

\texttt{HW.original} returns an object of class \texttt{HWorig} (what the Fortran subroutine \texttt{REGTST} return).

\texttt{LMR} calculates the L-moment ratios of a distribution given its parameters.

\texttt{PEL} calculates the parameters of a distribution given its L-moments.

\texttt{SAMLMR} calculates the sample L-moments ratios of a data-set.

\texttt{SAMLMU} calculates the ‘unbiased’ sample L-moments ratios of a data-set.

\texttt{SAMPWM} calculates the sample probability weighted moments of a data-set.

\texttt{REGLMR} calculates regional weighted averages of the sample L-moments ratios.

\texttt{REGTST} calculates statistics useful in regional frequency analysis. 1) Discordancy measure, \texttt{d(i)}, for individual sites in a region. Large values might be used as a flag to indicate potential errors in the data at the site. "large" might be 3 for regions with 15 or more sites, but less (exact values in array \texttt{dc1}) for smaller regions. 2) Heterogeneity measures, \texttt{H(j)}, for a region based upon either: if \texttt{j=1}: the weighted s.d. of the l-cv vs. l-skewness \texttt{j=3}: the average distance from the site to the regional average on a graph of l-skewness vs. l-kurtosis. In practice \texttt{H(1)} is probably sufficient. a value greater than (say) 1.0 suggests that further subdivision of the region should be considered as it might improve quantile estimates. 3) Goodness-of-fit measures, \texttt{Z(k)}, for 5 candidate distributions: \texttt{k=1}: generalized logistic \texttt{k=2}: generalized extreme value \texttt{k=3}: generalized normal (lognormal) \texttt{k=4}: pearson type iii (3-parameter gamma) \texttt{k=5}: generalized pareto. Provided that the region is acceptably close to homogeneous, the fit may be judged acceptable at 10 if \texttt{Z(k)} is less than 1.645 in absolute value.

For further details see Hosking and Wallis (1997), "Regional frequency analysis: an approach based on L-moments", cambridge university press, chapters 3-5.

**Note**

For information on the package and the Author, and for all the references, see \texttt{nsRFA}.
See Also

HW.tests.

Examples

data(hydroSIMN)
annualflows
summary(annualflows)
x <- annualflows["dato"],
cod <- annualflows["cod"],
split(x,cod)

HW.original(x,cod)

fac <- factor(annualflows["cod"],levels=c(34:38))
x2 <- annualflows[!is.na(fac),"dato"]
cod2 <- annualflows[!is.na(fac),"cod"]

HW.original(x2,cod2)

plot(HW.original(x2,cod2))

---

hydroSIMN Data-sample

Description

SIMN (Servizio Idrografico e Mareografico Nazionale) flow data samples and catchment parameters.

Usage

data(hydroSIMN)

Format

Data.frames:

annualflows is the data.frame of the annual flows with 3 columns: cod, the code of the station; anno, the year; dato, the value of the annual flow [mm].

parameters is the data.frame of parameters of 47 catchments with 16 columns: cod, the code of the station; Dm, the mean annual streamflow [mm] as reported in the ‘Pubblicazione n. 17’; Am, the mean annual rainfall [mm] as reported in the ‘Pubblicazione n. 17’; S, area of the plane projection of the drainage basin [km2]; Hm, mean elevation of the drainage basin [m a.s.l.]; Pm, mean slope of the basin [%]:

\[ P_m = \arctg(2(H_{med} - H_{min})/\sqrt{S}) \]

where \( S \) is the basin area, \( H_{med} \) the median elevation and \( H_{min} \) the elevation of the closing section. \( P_m \) is a slope measure of a square equivalent basin, and does not account for basin shape; LLDP,
length of the longest drainage path [km]. The longest drainage path is the longest path between the basin outlet and the most distant point on the basin border, following drainage directions. Actually the longest drainage path corresponds to the main stream plus the path on the hillslope that connects the stream source to the basin border; PLDP, slope of the longest drainage path [%]. Average of the slope values associated to each pixel in the longest drainage path; S2000, area above 2000 m a.s.l. [%]; EST, ‘easting’, sine of the angle between the segment connecting the center of mass and the outlet of the basin and the north. EST is 1 if the basin is oriented eastward, -1 if it is oriented westward; NORD, ‘northing’, cosine of the angle between the segment connecting the center of mass and the outlet of the basin and the north. NORD is 1 if the basin is oriented northward, -1 if it is oriented southward; Rc, circularity ratio Rc. Ratio between the basin area and the area of a circle having the same perimeter:

\[ R_c = \frac{4\pi S}{P^2} \]

where \( P \) is the watershed perimeter; xbar, longitude [deg] of the centroid of the plane projection of the drainage basin; ybar, latitude [deg] of the centroid of the plane projection of the drainage basin; IT, Thornthwaite index: a global moisture index that can be estimated, in its simplest form, as the ratio

\[ I_T = \frac{A_m - ET_0}{ET_0} \]

where \( ET_0 \) is the mean annual potential evapotranspiration on the basin; IB, Budyko index: a radiational aridity index expressed as

\[ I_B = \frac{R_n}{\lambda A_m} \]

where \( R_n \) is the mean annual net radiation and \( \lambda \) is the latent vaporization heat. Values assumed by \( I_B \) are lower than 1 for humid regions and greater than 1 in arid regions.

meanmonthlyflows is the data.frame of the mean monthly streamflows [mm] as reported in the ‘Pubblicazione n. 17’. It has 13 columns because the first one, cod, is the code of the station.

monthlyflows is the data.frame of the monthly streamflows [mm] with 4 columns: cod, the code of the station; anno, the year; mese, the month; dato, the value of the annual flow [mm].

Note

For information on the package and the Author, and for all the references, see nsRFA.

Examples

data(hydroSIMN)
annualflows
summary(annualflows)
x <- annualflows["dato"][,]
cod <- annualflows["cod"][,]
split(x,cod)
sapply(split(x,cod),mean)
sapply(split(x,cod),median)
sapply(split(x,cod),quantile)
sapply(split(x,cod),Lmoments)
parameters
KAPPA

Four parameter kappa distribution and L-moments

Description

KAPPA provides the link between L-moments of a sample and the four parameter kappa distribution.

Usage

f.kappa (x, xi, alfa, k, h)
F.kappa (x, xi, alfa, k, h)
invF.kappa (F, xi, alfa, k, h)
Lmom.kappa (xi, alfa, k, h)
par.kappa (lambda1, lambda2, tau3, tau4)
rand.kappa (numerosita, xi, alfa, k, h)

Arguments

x vector of quantiles
xi vector of kappa location parameters
alfa vector of kappa scale parameters
k vector of kappa third parameters
h vector of kappa fourth parameters
F vector of probabilities
lambda1 vector of sample means
lambda2 vector of L-variances
tau3 vector of L-CA (or L-skewness)
tau4 vector of L-kurtosis
numerosita numeric value indicating the length of the vector to be generated

Details

Definition

Parameters (4): $\xi$ (location), $\alpha$ (scale), $k$, $h$.

Range of $x$: upper bound is $\xi + \alpha/k$ if $k > 0$, $\infty$ if $k \leq 0$; lower bound is $\xi + \alpha(1 - h^{-k})/k$ if $h > 0$, $\xi + \alpha/k$ if $h \leq 0$ and $k < 0$ and $-\infty$ if $h \leq 0$ and $k \geq 0$

Probability density function:

$$f(x) = \alpha^{-1}\left[1 - k(x - \xi)/\alpha\right]^{1/k - 1}[F(x)]^{1-h}$$

Cumulative distribution function:

$$F(x) = \left\{1 - h\left[1 - k(x - \xi)/\alpha\right]^{1/k}\right\}^{1/h}$$
Quantile function:

\[ x(F) = \xi + \frac{\alpha}{k} \left[ 1 - \left( \frac{1 - F^h}{h} \right)^k \right] \]

\( h = -1 \) is the generalized logistic distribution; \( h = 0 \) is the generalized extreme value distribution; \( h = 1 \) is the generalized Pareto distribution.

**L-moments**

L-moments are defined for \( h \geq 0 \) and \( k > -1 \), or if \( h < 0 \) and \( -1 < k < -1/h \).

\[ \lambda_1 = \xi + \alpha(1 - g_1)/k \]
\[ \lambda_2 = \alpha(g_1 - g_2)/k \]
\[ \tau_3 = (-g_1 + 3g_2 - 2g_3)/(g_1 - g_2) \]
\[ \tau_4 = (-g_1 + 6g_2 - 10g_3 + 5g_4)/(g_1 - g_2) \]

where \( g_r = \frac{r\Gamma(1+k)\Gamma(r/h)}{h^{r+k}\Gamma(1+k+r/h)} \) if \( h > 0 \); \( g_r = \frac{r\Gamma(1+k)\Gamma(-k-r/h)}{(-h)^{r+k}\Gamma(1-r/h)} \) if \( h < 0 \).

Here \( \Gamma \) denote the gamma function

\[ \Gamma(x) = \int_0^\infty t^{x-1}e^{-t}dt \]

**Parameters**

There are no simple expressions for the parameters in terms of the L-moments. However they can be obtained with a numerical algorithm considering the formulations of \( \tau_3 \) and \( \tau_4 \) in terms of \( k \) and \( h \). Here we use the function \texttt{optim} to minimize \((t_3 - \tau_3)^2 + (t_4 - \tau_4)^2\) where \( t_3 \) and \( t_4 \) are the sample L-moment ratios.

\texttt{Lmom.kappa} and \texttt{par.kappa} accept input as vectors of equal length. In \texttt{f.kappa}, \texttt{F.kappa}, \texttt{invF.kappa} and \texttt{rand.kappa} parameters \((x_i, \alpha, k, h)\) must be atomic.

**Value**

\texttt{f.kappa} gives the density \( f \), \texttt{F.kappa} gives the distribution function \( F \), \texttt{invF.kappa} gives the quantile function \( x \), \texttt{Lmom.kappa} gives the L-moments \((\lambda_1, \lambda_2, \tau_3, \tau_4)\), \texttt{par.kappa} gives the parameters \((x_i, \alpha, k, h)\), and \texttt{rand.kappa} generates random deviates.

**Note**

For information on the package and the Author, and for all the references, see \texttt{nsRFA}.

**See Also**

\texttt{rnorm, runif, EXP, GENLOGIS, GENPAR, GEV, GUMBEL, LOGNORM, P3, optim, DISTPLOTS, GOFmontecarlo, Lmoments}.
Examples

```r
data(hydroSIMN)
annualflows
summary(annualflows)
x <- annualflows["dato"][,]
fac <- factor(annualflows["cod"][,])
split(x,fac)

camp <- split(x,fac)$"45"
ll <- Lmoments(camp)
parameters <- par.kappa(ll[1],ll[2],ll[4],ll[5])
f.kappa(1800,parameters$x,parameters$alfa,parameters$k,parameters$h)
F.kappa(1800,parameters$x,parameters$alfa,parameters$k,parameters$h)
invF.kappa(0.771088,parameters$x,parameters$alfa,parameters$k,parameters$h)
Lmom.kappa(parameters$x,parameters$alfa,parameters$k,parameters$h)
rand.kappa(100,parameters$x,parameters$alfa,parameters$k,parameters$h)

Rll <- regionalLmoments(x,fac); Rll
parameters <- par.kappa(Rll[1],Rll[2],Rll[4],Rll[5])
Lmom.kappa(parameters$x,parameters$alfa,parameters$k,parameters$h)
```

Description

Lmoments provides the estimate of L-moments of a sample or regional L-moments of a region.

Usage

- `Lmoments(x)`
- `regionalLmoments(x,cod)`
- `LCV(x)`
- `LCA(x)`
- `Lkur(x)`

Arguments

- `x` vector representing a data-sample (or data from many samples defined with `cod` in the case of `regionalLmoments`)
- `cod` array that defines the data subdivision among sites

Details

The estimation of L-moments is based on a sample of size \( n \), arranged in ascending order. Let \( x_{1:n} \leq x_{2:n} \leq \ldots \leq x_{n:n} \) be the ordered sample. An unbiased estimator of the probability
The sample L-moments are defined by:

\[ l_1 = b_0 \]
\[ l_2 = 2b_1 - b_0 \]
\[ l_3 = 6b_2 - 6b_1 + b_0 \]
\[ l_4 = 20b_3 - 30b_2 + 12b_1 - b_0 \]

and in general

\[ l_{r+1} = \sum_{k=0}^{r} \frac{(-1)^{r-k}(r+k)!}{(k!)^2(r-k)!} b_k \]

where \( r = 0, 1, \ldots, n - 1 \).

The sample L-moment ratios are defined by

\[ t_r = l_r/l_2 \]

and the sample L-CV by

\[ t = l_2/l_1 \]

Sample regional L-CV, L-skewness and L-kurtosis coefficients are defined as

\[ t^R = \frac{\sum_{i=1}^{k} n_i l_{i}^{(i)}}{\sum_{i=1}^{k} n_i} \]
\[ t_{3}^{R} = \frac{\sum_{i=1}^{k} n_i l_{3}^{(i)}}{\sum_{i=1}^{k} n_i} \]
\[ t_{4}^{R} = \frac{\sum_{i=1}^{k} n_i l_{4}^{(i)}}{\sum_{i=1}^{k} n_i} \]

Value

Lmoments gives the L-moments \( (l_1, l_2, t, t_3, t_4) \), regionalLmoments gives the regional weighted L-moments \( (l_1^R, l_2^R, t^R, t_3^R, t_4^R) \), LCV gives the coefficient of L-variation, LCA gives the L-skewness and Lkur gives the L-kurtosis of \( x \).

Note

For information on the package and the Author, and for all the references, see nsRFA.

See Also

mean, var, sd, HOMTESTS.
LOGNORM

Three parameter lognormal distribution and L-moments

Description

LOGNORM provides the link between L-moments of a sample and the three parameter log-normal distribution.

Usage

f.lognorm (x, xi, alfa, k)
F.lognorm (x, xi, alfa, k)
invF.lognorm (F, xi, alfa, k)
Lmom.lognorm (xi, alfa, k)
par.lognorm (lambda1, lambda2, tau3)
rand.lognorm (numerosita, xi, alfa, k)

Arguments

x       vector of quantiles
xi      vector of lognorm location parameters
alfa    vector of lognorm scale parameters
k       vector of lognorm shape parameters
F       vector of probabilities
lambda1 vector of sample means
lambda2 vector of L-variances
tau3    vector of L-CA (or L-skewness)
umerosita numeric value indicating the length of the vector to be generated
Details


Definition

Parameters (3): $\xi$ (location), $\alpha$ (scale), $k$ (shape).

Range of $x$: $-\infty < x \leq \xi + \alpha/k$ if $k > 0$; $-\infty < x < \infty$ if $k = 0$; $\xi + \alpha/k \leq x < \infty$ if $k < 0$.

Probability density function:

$$f(x) = \frac{e^{ky-y^2/2}}{\alpha\sqrt{2\pi}}$$

where $y = -k^{-1} \log\{1 - k(x - \xi)/\alpha\}$ if $k \neq 0$, $y = (x - \xi)/\alpha$ if $k = 0$.

Cumulative distribution function:

$$F(x) = \Phi(x)$$

where $\Phi(x) = \int_{-\infty}^{x} \phi(t)dt$.

Quantile function: $x(F)$ has no explicit analytical form.

$k = 0$ is the Normal distribution with parameters $\xi$ and $\alpha$.

L-moments

L-moments are defined for all values of $k$.

$$\lambda_1 = \xi + \alpha(1 - e^{k^2/2})/k$$

$$\lambda_2 = \alpha/ke^{k^2/2}[1 - 2\Phi(-k/\sqrt{2})]$$

There are no simple expressions for the L-moment ratios $\tau_r$ with $r \geq 3$. Here we use the rational-function approximation given in Hosking and Wallis (1997, p. 199).

Parameters

The shape parameter $k$ is a function of $\tau_3$ alone. No explicit solution is possible. Here we use the approximation given in Hosking and Wallis (1997, p. 199).

Given $k$, the other parameters are given by

$$\alpha = \frac{\lambda_2 ke^{-k^2/2}}{1 - 2\Phi(-k/\sqrt{2})}$$

$$\xi = \lambda_1 - \frac{\alpha}{k}(1 - e^{k^2/2})$$

Lmom.lognorm and par.lognorm accept input as vectors of equal length. In f.lognorm, F.lognorm, invF.lognorm and rand.lognorm parameters $(x_i, \alpha, k)$ must be atomic.

Value

f.lognorm gives the density $f$, F.lognorm gives the distribution function $F$, invF.lognorm gives the quantile function $x$, Lmom.lognorm gives the L-moments $(\lambda_1, \lambda_2, \tau_3, \tau_4)$, par.lognorm gives the parameters $(x_i, \alpha, k)$, and rand.lognorm generates random deviates.
Note

For information on the package and the Author, and for all the references, see nsRFA.

See Also

rnorm, runif, EXP, GENLOGIS, GENPAR, GEV, GUMBEL, KAPPA, P3, DISTPLOTS, GOFmontecarlo, Lmoments.

Examples

data(hydroSIMN)
annualflows
summary(annualflows)
x <- annualflows[,]['dato'][,]
fac <- factor(annualflows[,]['cod'][,])
split(x,fac)

camp <- split(x,fac)$"45"
ll <- Lmoments(camp)
parameters <- par.lognorm(ll[1],ll[2],ll[4])
f.lognorm(1800,parameters$x,parameters$alpha,parameters$k)
F.lognorm(1800,parameters$x,parameters$alpha,parameters$k)
invF.lognorm(0.7529877,parameters$x,parameters$alpha,parameters$k)
Lmom.lognorm(parameters$x,parameters$alpha,parameters$k)
rand.lognorm(100,parameters$x,parameters$alpha,parameters$k)

Rll <- regionalLmoments(x,fac); Rll
parameters <- par.lognorm(Rll[1],Rll[2],Rll[4])
Lmom.lognorm(parameters$x,parameters$alpha,parameters$k)

ML_estimation (x, dist="NORM")
moment_estimation (x, dist="NORM")

Description


Usage

ML_estimation (x, dist="NORM")
moment_estimation (x, dist="NORM")
moments

Arguments

x  data sample
dist  distribution: normal "NORM", Gumbel "GUMBEL", Generalized Extreme Value "GEV", Pearson type III "P3" and, only for sample_generator, Exponential "EXP"

Value

ML_estimation estimate the parameters of the distribution dist from a sample x using the maximum likelihood approach.
moment_estimation estimate the parameters of the distribution dist from a sample x using the moment method.

Note

For information on the package and the Author, and for all the references, see nsRFA.

See Also

GOFlaio2004.

Examples

# sample from an EV1 distribution
sm <- rand.gumb(100, 0, 1)
moment_estimation (sm, dist="GEV")
ML_estimation (sm, dist="GEV")

F.GEV(sm, -0.051, 0.97, -0.024)
rand.GEV (100, -0.051, 0.97, -0.024)
moment_estimation (sm, dist="P3")
ML_estimation (sm, dist="P3")

moments  Sample moments

Description

moments provides the estimate of the first 4 moment-statistics of a sample.

Usage

moments (x)
CV (x)
skew (x)
kurt (x)
Arguments

x vector representing a data-sample

Details

Skewness and kurtosis are defined as:

\[
\text{skew} = n^{-1} \sum_{i=1}^{n} \frac{(x_i - \text{mean}(x))^3}{sd(x)^3}
\]

\[
\text{kurt} = n^{-1} \sum_{i=1}^{n} \frac{(x_i - \text{mean}(x))^4}{sd(x)^4} - 3
\]


Note

For information on the package and the Author, and for all the references, see nsRFA.

See Also

mean, var, sd, Lmoments.

Examples

```r
x <- rnorm(30, 10, 2)
moments(x)

data(hydroSIMN)
x <- annualflows["dato"][,]
cod <- annualflows["cod"][,]
sapply(split(x, cod), moments)
```

---

MSClaio2008 Model Selection Criteria

Description


Usage

```r
MSClaio2008 (sample, dist=c("NORM", "LN", "GUMBEL", "EV2", "GEV", "P3", "LP3"),
          crit=c("AIC", "AICc", "BIC", "ADC"))
## S3 method for class 'MSClaio2008'
print(x, digits=max(3,getOption("digits") - 3), ...)
## S3 method for class 'MSClaio2008'
summary(object, ...)
## S3 method for class 'MSClaio2008'
plot(x, ...)
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sample</td>
<td>data sample</td>
</tr>
<tr>
<td>crit</td>
<td>Model-selection criteria: Akaike Information Criterion &quot;AIC&quot;, Akaike Information Criterion corrected &quot;AICc&quot;, Bayesian Information Criterion &quot;BIC&quot;, Anderson-Darling Criterion &quot;ADC&quot;</td>
</tr>
<tr>
<td>x</td>
<td>object of class MSClaio2008, output of MSClaio2008()</td>
</tr>
<tr>
<td>object</td>
<td>object of class MSClaio2008, output of MSClaio2008()</td>
</tr>
<tr>
<td>digits</td>
<td>minimal number of &quot;significant&quot; digits, see 'print.default'</td>
</tr>
<tr>
<td>...</td>
<td>other arguments</td>
</tr>
</tbody>
</table>

Details

The following lines are extracted from Laio et al. (2008). See the paper for more details and references.

Model selection criteria

The problem of model selection can be formalized as follows: a sample of \( n \) data, \( D = (x_1, \ldots, x_n) \), arranged in ascending order is available, sampled from an unknown parent distribution \( f(x) \); \( N_m \) operating models, \( M_j, j = 1, \ldots, N_m \), are used to represent the data. The operating models are in the form of probability distributions, \( M_j = g_j(x, \hat{\theta}) \), with parameters \( \hat{\theta} \) estimated from the available data sample \( D \). The scope of model selection is to identify the model \( M_{\text{opt}} \) which is better suited to represent the data, i.e. the model which is closer in some sense to the parent distribution \( f(x) \).

Three different model selection criteria are considered here, namely, the Akaike Information Criterion (AIC), the Bayesian Information Criterion (BIC), and the Anderson-Darling Criterion (ADC).

Akaike Information Criterion

The Akaike Information Criterion (AIC) for the j-th operational model can be computed as

\[
AIC_j = -2\ln(L_j(\hat{\theta})) + 2p_j
\]

where

\[
L_j(\hat{\theta}) = \prod_{i=1}^{n} g_j(x_i, \hat{\theta})
\]

is the likelihood function, evaluated at the point \( \theta = \hat{\theta} \) corresponding to the maximum likelihood estimator of the parameter vector \( \theta \) and \( p_j \) is the number of estimated parameter of the j-th operational model. In practice, after the computation of the \( AIC_j \), for all of the operating models, one selects the model with the minimum AIC value, \( AIC_{\text{min}} \).

When the sample size, \( n \), is small, with respect to the number of estimated parameters, \( p \), the AIC may perform inadequately. In those cases a second-order variant of AIC, called AICc, should be used:

\[
AICc_j = -2\ln(L_j(\hat{\theta})) + 2p_j(n/(n-p_j-1))
\]
Indicatively, AICc should be used when \(n/p < 40\).

**Bayesian Information Criterion**

The Bayesian Information Criterion (BIC) for the \(j\)-th operational model reads

\[
BIC_j = -2\ln(L_j(\hat{\theta})) + \ln(n)p_j
\]

In practical application, after the computation of the \(BIC_j\), for all of the operating models, one selects the model with the minimum BIC value, \(BIC_{\text{min}}\).

**Anderson-Darling Criterion**

The Anderson-Darling criterion has the form:

\[
ADC_j = 0.0403 + 0.116((\Delta_{AD,j} - \epsilon_j)/\beta_j)^{(\eta_j/0.851)}
\]

if \(1.2\epsilon_j < \Delta_{AD,j}\),

\[
ADC_j = [0.0403 + 0.116((0.2\epsilon_j)/\beta_j)^{(\eta_j/0.851)}](\Delta_{AD,j} - 0.2\epsilon_j/\epsilon_j)
\]

if \(1.2\epsilon_j \geq \Delta_{AD,j}\), where \(\Delta_{AD,j}\) is the discrepancy measure characterizing the criterion, the Anderson-Darling statistic \(A^2\) in \texttt{GOFlaio2004}, and \(\epsilon_j\), \(\beta_j\) and \(\eta_j\) are distribution-dependent coefficients that are tabled by Laio [2004, Tables 3 and 5] for a set of seven distributions commonly employed for the frequency analysis of extreme events. In practice, after the computation of the \(ADC_j\), for all of the operating models, one selects the model with the minimum ADC value, \(ADC_{\text{min}}\).

**Value**

\texttt{MSClaio2008} returns the value of the criteria \texttt{crit} (see Details) chosen applied to the sample, for every distribution \texttt{dist}.

\texttt{plot.MSClaio2008} plots the empirical distribution function of sample (Weibull plotting position) on a log-normal probability plot, plots the candidate distributions \texttt{dist} (whose parameters are evaluated with the maximum likelihood technique, see \texttt{MLlaio2004}, and highlights the ones chosen by the criteria \texttt{crit}).

**Note**

For information on the package and the Author, and for all the references, see \texttt{nsRFA}.

**See Also**

\texttt{GOFlaio2004,MLlaio2004}.

**Examples**

```r
data(FEH1000)
sitedata <- am[am[,1]==53004,] # data of site 53004
serieplot(sitedata[,4], sitedata[,3])
MSC <- MSClaio2008(sitedata[,4])
MSC
summary(MSC)
```
plot(MSC)

sitedata <- am[am[,1]==69023,] # data of site 69023
serieplot(sitedata[,4], sitedata[,3])
MSC <- MS Claio2008(sitedata[,4], crit=c("AIC", "ADC"))
MSC
summary(MSC)
plot(MSC)

sitedata <- am[am[,1]==83802,] # data of site 83802
serieplot(sitedata[,4], sitedata[,3])
MSC <- MS Claio2008(sitedata[,4], dist=c("GEV", "P3", "LP3"))
MSC
summary(MSC)
plot(MSC)

# short sample, high positive L-CA
sitedata <- am[am[,1]==40012,] # data of site 40012
serieplot(sitedata[,4], sitedata[,3])
MSC <- MS Claio2008(sitedata[,4])
MSC
summary(MSC)
plot(MSC)

# negative L-CA
sitedata <- am[am[,1]==68002,] # data of site 68002
serieplot(sitedata[,4], sitedata[,3])
MSC <- MS Claio2008(sitedata[,4])
MSC
summary(MSC)
plot(MSC)

---

P3

Three parameters Pearson type III distribution and L-moments

Description

P3 provides the link between L-moments of a sample and the three parameter Pearson type III distribution.

Usage

f.gamma (x, xi, beta, alfa)
F.gamma (x, xi, beta, alfa)
invF.gamma (F, xi, beta, alfa)
Lmom.gamma (xi, beta, alfa)
par.gamma (lambda1, lambda2, tau3)
rand.gamma (numerosita, xi, beta, alfa)
mom2par.gamma (mu, sigma, gamm)
par2mom.gamma (alfa, beta, xi)
Arguments

- **x**: vector of quantiles
- **mu**: vector of gamma mean
- **sigma**: vector of gamma standard deviation
- **gamm**: vector of gamma third moment
- **F**: vector of probabilities
- **lambda1**: vector of sample means
- **lambda2**: vector of L-variances
- **tau3**: vector of L-CA (or L-skewness)
- **numerosita**: numeric value indicating the length of the vector to be generated
- **alfa**: vector of gamma shape parameters
- **beta**: vector of gamma scale parameters
- **xi**: vector of gamma location parameters

Details


**Definition**

Parameters (3): \( \xi \) (location), \( \beta \) (scale), \( \alpha \) (shape). Moments (3): \( \mu \) (mean), \( \sigma \) (standard deviation), \( \gamma \) (skewness).

If \( \gamma \neq 0 \), let \( \alpha = 4/\gamma^2 \), \( \beta = 1/2|\gamma| \), and \( \xi = \mu - 2\sigma/\gamma \). If \( \gamma > 0 \), then the range of \( x \) is \( \xi \leq x < \infty \) and

\[
    f(x) = \frac{(x-\xi)^{\alpha-1}e^{-(x-\xi)/\beta}}{\beta^\alpha \Gamma(\alpha)}
\]

\[
    F(x) = G\left(\alpha, \frac{x-\xi}{\beta}\right)/\Gamma(\alpha)
\]

If \( \gamma = 0 \), then the distribution is Normal, the range of \( x \) is \( -\infty < x < \infty \) and

\[
    f(x) = \phi\left(\frac{x-\mu}{\sigma}\right)
\]

\[
    F(x) = \Phi\left(\frac{x-\mu}{\sigma}\right)
\]

where \( \phi(x) = (2\pi)^{-1/2} \exp\left(-x^2/2\right) \) and \( \Phi(x) = \int_{-\infty}^{x} \phi(t)dt \).

If \( \gamma < 0 \), then the range of \( x \) is \( -\infty < x \leq \xi \) and

\[
    f(x) = \frac{(\xi-x)^{\alpha-1}e^{-(\xi-x)/\beta}}{\beta^\alpha \Gamma(\alpha)}
\]
\[ F(x) = G \left( \alpha, \frac{\xi - x}{\beta} \right) / \Gamma(\alpha) \]

In each case, \( x(F) \) has no explicit analytical form. Here \( \Gamma \) is the gamma function, defined as

\[ \Gamma(x) = \int_0^\infty t^{x-1}e^{-t}dt \]

and

\[ G(\alpha, x) = \int_0^x t^{\alpha-1}e^{-t}dt \]

is the incomplete gamma function.

\( \gamma = 2 \) is the exponential distribution; \( \gamma = 0 \) is the Normal distribution; \( \gamma = -2 \) is the reverse exponential distribution.

The parameters \( \mu, \sigma \) and \( \gamma \) are the conventional moments of the distribution.

**L-moments**

Assuming \( \gamma > 0 \), L-moments are defined for \( 0 < \alpha < \infty \).

\[ \lambda_1 = \xi + \alpha\beta \]

\[ \lambda_2 = \pi^{-1/2}\beta\Gamma(\alpha + 1/2)/\Gamma(\alpha) \]

\[ \tau_3 = 6I_{1/\beta}(\alpha, 2\alpha) - 3 \]

where \( I_x(p, q) \) is the incomplete beta function ratio

\[ I_x(p, q) = \frac{\Gamma(p + q)}{\Gamma(p)\Gamma(q)} \int_0^x t^{p-1}(1 - t)^{q-1}dt \]

There is no simple expression for \( \tau_4 \). Here we use the rational-fucnion approximation given by Hosking and Wallis (1997, pp. 201-202).

The corresponding results for \( \gamma < 0 \) are obtained by changing the signs of \( \lambda_1, \tau_3 \) and \( \xi \) wherever they occur above.

**Parameters**

\( \alpha \) is obtained with an approximation. If \( 0 < |\tau_3| < 1/3 \), let \( z = 3\pi\tau_3^2 \) and use

\[ \alpha \approx \frac{1 + 0.2906z}{z + 0.1882z^2 + 0.0442z^3} \]

if \( 1/3 < |\tau_3| < 1 \), let \( z = 1 - |\tau_3| \) and use

\[ \alpha \approx \frac{0.36067z - 0.59567z^2 + 0.25361z^3}{1 - 2.78861z + 2.56096z^2 - 0.77045z^3} \]

Given \( \alpha \), then \( \gamma = 2\alpha^{-1/2}\text{sign}(\tau_3) \), \( \sigma = \lambda_2\pi^{1/2}\alpha^{1/2}\Gamma(\alpha)/\Gamma(\alpha + 1/2) \), \( \mu = \lambda_1 \).

lmom.gamma and par.gamma accept input as vectors of equal length. In f.gamma, F.gamma, invF.gamma and rand.gamma parameters (mu, sigma, gamm) must be atomic.
Value

f.gamma gives the density \( f \), F.gamma gives the distribution function \( F \), invF.gamma gives the quantile function \( x \). Lmom.gamma gives the L-moments \( (\lambda_1, \lambda_2, \tau_3, \tau_4) \), par.gamma gives the parameters \( (\mu, \sigma, \gamma) \), and rand.gamma generates random deviates.

mom2par.gamma returns the parameters \( \alpha, \beta \) and \( \xi \), given the parameters (moments) \( \mu, \sigma, \gamma \).

Note

For information on the package and the Author, and for all the references, see nsRFA.

See Also

rnorm, runif, EXP, GENLOGIS, GENPAR, GEV, GUMBEL, KAPPA, LOGNORM; DISTPLOTS, GOFmontecarlo, Lmoments.

Examples

data(hydroSIMN)
annualflows
summary(annualflows)
x <- annualflows["dato"]
fac <- factor(annualflows["cod"])
split(x,fac)
camp <- split(x,fac)$"45"
ll <- Lmoments(camp)
parameters <- par.gamma(ll[1],ll[2],ll[4])
f.gamma(1800,parameters$xi,parameters$beta,parameters$alfa)
F.gamma(1800,parameters$xi,parameters$beta,parameters$alfa)
invF.gamma(0.7511627,parameters$xi,parameters$beta,parameters$alfa)
Lmom.gamma(parameters$xi,parameters$beta,parameters$alfa)
rand.gamma(100,parameters$xi,parameters$beta,parameters$alfa)
Rl1 <- regionallmoments(x,fac); Rl1
parameters <- par.gamma(Rl1[1],Rl1[2],Rl1[4])
Lmom.gamma(parameters$xi,parameters$beta,parameters$alfa)

moments <- par2mom.gamma(parameters$alfa,parameters$beta,parameters$xi); moments
mom2par.gamma(moments$mu,moments$sigma,moments$xi)

Description

Diagnostics of the output of \code{lm}, that is used to fit linear models.
Usage

\begin{align*}
R2.lm \ (x) \\
prt.lm \ (x) \\
mantel.lm \ (x, \ Nperm = 1000) \\
vif.lm \ (x) \\
RMSE.lm \ (x) \\
MAE.lm \ (x) \\
predinterval.lm \ (x, \ level = 0.95) \\
jackknife1.lm \ (x) \\
RMSEjk.lm \ (x) \\
MAEjk.lm \ (x)
\end{align*}

Arguments

- **x**: object of class "lm" (output of ‘lm’)
- **Nperm**: number of permutations
- **level**: significance level

Details

`mantel.lm` is performed under the assumption that the dependent distance matrix is variable, while the independent distance matrices are fixed and measured without error (they are not related to random variables, see Smouse et al., 1986). Under this assumption, the significance of the regression between distance matrices can be evaluated simultaneously permuting the rows and corresponding columns in only the dependent distance matrix, while the others are held constant.

Value

- **R2.lm** returns the coefficient of determination $R^2$ and the adjusted coefficient of determination $R^2_{adj}$ of the regression.
- **prt.lm** returns the probability $Pr(|t|)$ of the significance test (Student t) of the independent variables. If the value is 0.06 for a regressor, its coefficient is not significantly different from 0 for a test with significance level of 5%.
- **mantel.lm** returns the probability $P$ of the Mantel test on every variable conditioned to the others. It substitutes **prt.lm** when dealing with distance matrices. If $P$ is, for example, 0.92, the variable should be considered significant with significance level greater of 8%.
- **vif.lm** returns the variance inflation factors (VIF) of the independent values of the regression. If $VIF > 5$ (or 10) there is a problem of multicollinearity.
- **RMSE.lm** returns the root mean squared error of the regression.
- **MAE.lm** returns the mean absolute error of the regression.
- **predinterval.lm** returns the prediction intervals at a specified `level` in correspondence to the fitted data.
- **jackknife1.lm** returns predicted values by a jackknife (cross-validation) procedure. The procedure (remove 1 observation, fit the model, estimate in the removed point) is repeated for all the points.
**RMSEjk.lm** returns the root mean squared error of the cross-validation (performed with jackknife1.lm).

**MAEjk.lm** returns the mean absolute error of the cross-validation (performed with jackknife1.lm).

**Note**

For information on the package and the Author, and for all the references, see nsRFA.

**See Also**

lm, summary.lm, predict.lm

**Examples**

data(hydroSIMN)

D <- annualflows["dato"],
cod <- annualflows["cod"],

#Dm <- tapply(D,cod,mean)
#datregr <- cbind(Dm,parameters)

datregr <- parameters

tregr0 <- lm(Dm ~ .,datregr); summary(tregr0)
tregr1 <- lm(Dm ~ Am + Hm + Ybar,datregr); summary(tregr1)

R2.lm(tregr0)
R2.lm(tregr1)

trt.lm(tregr0)
trt.lm(tregr1)

vif.lm(tregr0)
vif.lm(tregr1)

RMSE.lm(tregr0)
RMSE.lm(tregr1)

MAE.lm(tregr0)
MAE.lm(tregr1)

predinterval.lm(tregr0)
jackknife1.lm(tregr0)
jackknife1.lm(tregr1)

RMSEjk.lm(tregr0)
RMSEjk.lm(tregr1)

MAEjk.lm(tregr0)
MAEjk.lm(tregr1)

# mantel test on distance matrices
#Y <- AD.dist(D,cod)         # it takes some time
#X <- data.frame(apply(datregr[,c("Hm","Ybar")],2,dist))
#dati <- cbind(X)
#modello <- lm(Y ~ Hm + Ybar, dati)
#mantel.lm(modello, Nperm=100)

---

**roi**

*Region of influence*

**Description**

Formation of clusters for Regional Frequency Analysis: region of influence (Burn, 1990).

**Usage**

```r
roi (p.ungauged, p.gauged, cod.p, x=NULL, cod=NULL)
roi.hom (p.ungauged, p.gauged, cod.p, x, cod, test="HW", limit=2, Nsim=500, index=2)
roi.st.year (p.ungauged, p.gauged, cod.p, x, cod, test="HW", station.year=500, Nsim=500, index=2)
```

**Arguments**

- `x` vector representing data from many samples defined with `cod`
- `cod` array that defines the data subdivision among sites
- `index` if `index=1` samples are divided by their average value; if `index=2` (default) samples are divided by their median value
- `p.ungauged` parameters of the ungauged site (1 row)
- `p.gauged` parameters of gauged sites
- `cod.p` code of gauged sites
- `test` homogeneity test to apply: "HW" (default) or "AD" (in `roi.st.year` you can choose "HW" and "AD" too)
- `limit` limit over which regions must be considered heterogeneous: for example 2 for "HW" or .95 for "AD"
- `Nsim` number of simulations in "HW" or "AD" tests
- `station.year` number of station years to form the region

**Details**

The Euclidean distance is used. Given `p` different classification variables, the distance between two elements `i` and `j` is:

\[
d_{ij} = \sqrt{\frac{1}{p} \sum_{h=1}^{p} (x_{hi} - x_{hj})^2}
\]

where `x_{hi}` is the value of the `h`-th variable of the `i`-th element.
roi returns the ‘region of influence’ for the site defined with p.ungauged. It the gauged sites ordered according to the euclidean distance against the site of interest (the distance is evaluated in the space defined by parameters p.ungauged and p.gauged). If x=NULL and cod=NULL (default), a data.frame with the ordered sites and the distances against the site of interest is returned. If x and cod are provided, the data.frame will contain also statistics of samples (number of data n and L-moments).

roi.hom returns the ‘region of influence’ for the site defined with p.ungauged. It returns codes of gauged sites that form an homogeneous region according to the Hosking and Wallis "HW" or Anderson-Darling "AD" tests. The region is formed using distances in the space defined by parameters p.ungauged and p.gauged.

roi.st.year returns the ‘region of influence’ for the site defined with p.ungauged. It returns codes of gauged sites that form a region and the risult of homogeneity tests, according to the station-year criterion. It also return the similarity ranking factor \( S_i \), the weights \( w_i \) and the regional L-moments as evaluated in the Flood Estimation Handbook (Robson and Reed, 1999). The region is formed using distances in the space defined by parameters p.ungauged and p.gauged.

Note

For information on the package and the Author, and for all the references, see nsRFA.

See Also

traceWminim, AD.dist, HOMTESTS for the definition of the Hosking and Wallis "HW" or Anderson-Darling "AD" tests.

Examples

data(hydroSIMN)
parameters
summary(parameters)

annualflows
summary(annualflows)
x <- annualflows["dato"],
cod <- annualflows["cod"],

roi(parameters[5,3:5],parameters[-5,3:5],parameters[-5,1])
roi(parameters[5,3:5],parameters[-5,3:5],parameters[-5,1],x,cod)

# roi.hom
#roi.hom(parameters[5,3:5],parameters[-5,3:5],parameters[-5,1],x,cod)  # it takes some time
#roi.hom(parameters[5,3:5],parameters[-5,3:5],parameters[-5,1],x,cod,   # test="AD",limit=.95)  # it takes some time

#roi.hom(parameters[8,3:5],parameters[-8,3:5],
#        parameters[-8,1],x,cod)  # it takes some time
SERIESPLOTS

SERIESPLOTS

Series plots

Description

Plots for time-series investigation.

Usage

serieplot (x, t, lim.x=c(min(x), max(x)), lim.t=c(min(t), max(t)),
...)
consistencyplot (t, cod, cex.axis=.8, mark.code=TRUE, ...)

Arguments

x data sample
t vector representing time (e.g. years) of data-samples defined with cod
lim.x, lim.t plot limits
cod array that defines the data subdivision among sites
cex.axis dimensions of points and labels
mark.code if TRUE (default) codes of samples are plotted on y axis
... graphical parameters as xlab, ylab, main, ...

Value

consistencyplot displays time-series consistency.

Note

For information on the package and the Author, and for all the references, see nsRFA.

See Also

plot, DISTPLOTS
Examples

```r
data(hydroSIMN)
annualflows[c(1:10),]
x <- annualflows["dato"][,]
y <- annualflows["anno"][,]
cod <- annualflows["cod"][,]
consistencyplot(y,cod)

for (i in unique(cod)) {
  serieplot(x[cod==i], y[cod==i], c(0,max(x)), c(min(y),max(y)),
    xlab="", ylab="D [mm]", main=i)
  readline()
}
```

---

**STATICPLOTS**

**Static plots**

**Description**

Plots from books and articles.

**Usage**

- `Lmoment.ratio.diagram(grid=TRUE, ...)`
- `Lspace.HWvsAD(grid=TRUE, ...)`
- `Lspace.limits(grid=TRUE, ...)`

**Arguments**

- `grid` should a grid be plotted?
- `...` other arguments

**Value**

`Lmoment.ratio.diagram` plots points corresponding to two parameters distributions and lines corresponding to three parameters distributions on the 'L-CA - L-kur' plane. The distributions are: E = exponential, G = gumbel, L = logistic, N = normal, U = uniform, GLO = generalized logistic, GEV = generalized extreme-value, GPA = generalized Pareto, LN3 = lognormal, PE3 = Pearson type III. `Lspace.HWvsAD` separate regions, in 'L-CA - L-CV' space, where the homogeneity tests of Hosking and Wallis (HW) and Anderson-Darling (AD) are preferable. `Lspace.limits` displays limits for regional L-moments in the 'L-CA - L-CV'.

**Note**

For information on the package and the Author, and for all the references, see `nsRFA`.

---

**STATICPLOTS**
traceWminim

See Also

EXP, GENLOGIS, GENPAR, LOGNORM, GUMBEL, GEV, P3

Examples

Lmoment.ratio.diagram()
Lspace.HWvsAD()
Lspace.limits()

data(hydroSIMN)
annualflows[c(1:10),]
x <- annualflows["dato",]
cod <- annualflows["cod",]
rlm <- regionalLmoments(x, cod)
Lmoment.ratio.diagram()
points(rlm["lcaR"], rlm["lkurR"], col = "red", pch = 19)
Lspace.HWvsAD()
points(rlm["lcaR"], rlm["lcvR"], col = "red", pch = 19)

traceWminim

Cluster analysis: disjoint regions

Description

Formation of disjoint regions for Regional Frequency Analysis.

Usage

traceWminim (X, centers)
sumtraceW (clusters, X)
nearest (clusters, X)

Arguments

X a numeric matrix of characteristics, or an object that can be coerced to such a matrix (such as a numeric vector or a data frame with all numeric columns)

centers the number of clusters

clusters a numeric vector containing the subdivision of X in clusters

Details

The Euclidean distance is used. Given \( p \) different classification variables, the distance between two elements \( i \) and \( j \) is:

\[
d_{ij} = \sqrt{\frac{1}{p} \sum_{h=1}^{p} (x_{hi} - x_{hj})^2}
\]
where \(x_{hi}\) is the value of the \(h\)-th variable of the \(i\)-th element.

The function `traceWminim` is a composition of a hierarchical algorithm, the Ward (1963) one, and an optimisation procedure consisting in the minimisation of:

\[
W = \sum_{i=1}^{k} \left( \sum_{j=1}^{n_i} \delta_{ij}^2 \right)
\]

where \(k\) is the number of clusters (obtained initially with Ward’s algorithm), \(n_i\) is the number of sites in the \(i\)-th cluster and \(\delta_{ij}\) is the Euclidean distance between the \(j\)-th element of the \(i\)-th group and the center of mass of the \(i\)-th cluster. \(W\) is calculated with `sumtraceW`. The algorithm consists in moving a site from one cluster to another if this makes \(W\) decrease.

**Value**

`traceWminim` gives a vector defining the subdivision of elements characterized by \(X\) in \(n=\text{centers}\) clusters.

`sumtraceW` gives \(W\) (it is used by `traceWminim`).

`nearest` gives the nearest site to the centers of mass of clusters (it is used by `traceWminim`).

**Note**

For information on the package and the Author, and for all the references, see `nsRFA`.

**See Also**

`roi`, `AD.dist`.

**Examples**

```r
data(hydroSIMN)
parameters
summary(parameters)

# traceWminim
param <- parameters[,c("Hm","Ybar")]
n <- dim(param)[1]; k <- dim(param)[2]
param.norm <- (param - matrix(apply(param,2,mean),nrow=n,ncol=k, byrow=TRUE))/matrix(apply(param,2,sd), nrow=n,ncol=k,byrow=TRUE)
clusters <- traceWminim(param.norm,4);
names(clusters) <- parameters[,"cod"]
clusters

annualflows
summary(annualflows)
x <- annualflows[,"dato"]
cod <- annualflows[,"cod"]
fac <- factor(annualflows[,"cod"],
              levels=names(clusters[clusters==1]))
```
varLmoments

Description

varLmoments provides distribution-free unbiased estimators of the variances and covariances of sample L-moments.

Usage

varLmoments (x, matrix=TRUE)
varLCV (x)
varLCA (x)
varLkur (x)

Arguments

x vector representing a data-sample
matrix if TRUE (default), the matrix of estimates of the variance structure (variance and covariance) i of sample L-moments is returned; if FALSE, a vector containing \( \text{var}(l_1) \), \( \text{var}(l_2) \), \( \text{var}(l_3) \), \( \text{var}(l_4) \), \( \text{var}(t) \), \( \text{var}(t_3) \) and \( \text{var}(t_4) \) is returned.

Details

The estimation of the exact variance structure of sample L-moments is based on Elamir et Seheult (2004).

Value

varLmoments gives the matrix of unbiased estimates of the variance structure of sample L-moments:
this is a 4x4 matrix containing \( \text{var}(l_1) \), \( \text{var}(l_2) \), \( \text{var}(l_3) \), \( \text{var}(l_4) \) on the main diagonal, and the correspondant covariances elsewhere (\( \text{cov}(l_1,l_2) \), \( \text{cov}(l_1,l_3) \), etc.):
varLCV gives the unbiased estimate of the variance of sample coefficient of L-variation of \( x \);
varLCA gives the unbiased estimate of the variance of sample L-skewness of \( x \);
varLkur gives the unbiased estimate of the variance of sample L-kurtosis of \( x \).
Note
For information on the package and the Author, and for all the references, see nsRFA.

See Also
var, Lmoments.

Examples
x <- rnorm(30,10,2)
varLmoments(x)
varLmoments(x, FALSE)

varLCV(x)
varLCA(x)
varLkur(x)

data(hydroSIMN)
x <- annualflows["dato"],
cod <- annualflows["cod"],
dvarLmom <- function(x) {diag(varLmoments(x))}
sapply(split(x,cod), dvarLmom)
Index

* cluster
  AD.dist, 8
  roi, 67
  traceWminim, 71
* datasets
  Ardechedata, 9
  FEH1000, 27
  functionsLaio, 28
  hydroSIMN, 48
* distribution
  EXP, 24
  GENLOGIS, 28
  GENPAR, 30
  GEV, 33
  GUMBEL, 39
  KAPPA, 50
  LOGNORM, 54
  nsRFA-package, 3
  P3, 61
* hplot
  DISTPLOTS, 6
  SERIESPLOTS, 69
  STATICPLOTS, 70
* htest
  BayesianMCMC, 10
  GOFlaio2004, 35
  GOFmontecarlo, 36
  HOMTESTS, 41
  HW.original, 45
  MLlaio2004, 56
  MSciaio2008, 58
  nsRFA-package, 3
* models
  DIAGNOSTICS, 19
* package
  nsRFA-package, 3
* regression
  bestlm, 17
  REGRDIAGNOSTICS, 64
* univar
  Lmoments, 52
  moments, 57
  nsRFA-package, 3
  varLmoments, 73

A2, 4
A2 (GOFlaio2004), 35
A2_GOFlaio (GOFlaio2004), 35
AD.dist, 4, 8, 68, 72
ADbootstrap.test, 4
ADbootstrap.test (HOMTESTS), 41
am (FEH1000), 27
annualflows (hydroSIMN), 48
Ardeche_areas (Ardechedata), 9
Ardeche_ungauged_extremes (Ardechedata), 9
Ardechedata, 9
BayesianMCMC, 5, 6, 10, 22
BayesianMCMCcont (BayesianMCMC), 10
BayesianMCMCreg, 6
BayesianMCMCreg (BayesianMCMC), 10
BayesianMCMCregcont (BayesianMCMC), 10
Beauvene_cont (Ardechedata), 9
bestlm, 4, 6, 17

cd (FEH1000), 27
Chambonas_cont (Ardechedata), 9
consistencyplot (SERIESPLOTS), 69
criticalD (HOMTESTS), 41
CV (moments), 57
df.kgev (functionsLaio), 28
df.polig (functionsLaio), 28
DIAGNOSTICS, 4, 19
discordancy (HOMTESTS), 41
dist, 4
DISTPLOTS, 6, 21, 26, 30, 32, 34, 40, 51, 56, 64, 69
DK.test (HOMTESTS), 41

EXP, 24, 30, 32, 34, 40, 51, 56, 64, 71
expplot (DISTPLOTS), 21
exppoints (DISTPLOTS), 21

F.exp (EXP), 24
f.exp (EXP), 24
F.gamma (P3), 61
f.gamma (P3), 61
F.genlogis (GENLOGIS), 28
f.genlogis (GENLOGIS), 28
F.genpar (GENPAR), 30
f.genpar (GENPAR), 30
F.GEV (GEV), 33
f.GEV (GEV), 33
F.gumb (GUMBEL), 39
f.gumb (GUMBEL), 39
F.kappa (KAPPA), 50
f.kappa (KAPPA), 50
F.lognorm (LOGNORM), 54
f.lognorm (LOGNORM), 54
FEH1000, 27
frechetplot (DISTPLOTS), 21
functionsLaio, 28
fw2 (GOFlaio2004), 35
gammaplot (DISTPLOTS), 21
gammapoints (DISTPLOTS), 21
genlogis, 26, 32, 34, 40, 51, 56, 64, 71
genpar, 26, 30, 32, 34, 40, 51, 56, 64, 71
gev, 26, 30, 32, 33, 40, 51, 56, 64, 71
gofEXPtest (GOFlmontecarlo), 36
gofGENLOGISTest (GOFlmontecarlo), 36
gofGENPARtest (GOFlmontecarlo), 36
gofGEVtest (GOFlmontecarlo), 36
gofGUMBELtest (GOFlmontecarlo), 36
GOFlaio2004, 5, 35, 57, 60
gofLOGNORMtest (GOFlmontecarlo), 36
GOFlmontecarlo, 5, 6, 26, 30, 32, 34, 36, 40, 51, 56, 64
gofNORMTest (GOFlmontecarlo), 36
gofP3test (GOFlmontecarlo), 36
GUMBEL, 23, 26, 30, 32, 34, 39, 51, 56, 64, 71
gumbelplot (DISTPLOTS), 21
gumbelpoints (DISTPLOTS), 21

HOMTESTS, 4, 38, 41, 53, 68
HW.original, 45

HW.tests, 4, 48
HW.tests (HOMTESTS), 41
hydroSIMN, 48

invF.exp (EXP), 24
invF.gamma (P3), 61
invF.genlogis (GENLOGIS), 28
invF.genpar (GENPAR), 30
invF.GEV (GEV), 33
invF.gumb (GUMBEL), 39
invF.kappa (KAPPA), 50
invF.lognorm (LOGNORM), 54

jackknife.lm (REGRDIGNOSTICS), 64

KAPPA, 26, 30, 32, 34, 40, 45, 50, 56, 64
kurt (moments), 57

LCA (Lmoments), 52
LCV (Lmoments), 52
Lkur (Lmoments), 52
lm, 3, 4, 18, 20, 66
Lmom.exp (EXP), 24
Lmom.gamma (P3), 61
Lmom.genlogis (GENLOGIS), 28
Lmom.genpar (GENPAR), 30
Lmom.GEV (GEV), 33
Lmom.gumb (GUMBEL), 39
Lmom.kappa (KAPPA), 50
Lmom.lognorm (LOGNORM), 54
Lmoment.ratio.diagram (STATICPLOTS), 70
Lmoments, 4, 26, 30, 32, 34, 40, 51, 52, 56, 58, 64, 74

LMR (HW.original), 45

logisplot (DISTPLOTS), 21
logispoints (DISTPLOTS), 21
loglogplot (DISTPLOTS), 21
loglogpoints (DISTPLOTS), 21
LOGNORM, 23, 26, 30, 32, 34, 40, 51, 54, 64, 71
Lognormal, 23
lognormplot (DISTPLOTS), 21
Lspace.HWvsAD (STATICPLOTS), 70
Lspace.limits (STATICPLOTS), 70

MAE (DIAGNOSTICS), 19
MAE.lm (REGRDIGNOSTICS), 64
MAEm.jk.lm (REGRDIGNOSTICS), 64
MAEP (DIAGNOSTICS), 19
mantel.lm, 4
INDEX

mantel.lm (REGRDIAGNOSTICS), 64
mean, 53, 58
meanmonthlyflows (hydroSIMN), 48
ML_estimation (MLlaio2004), 56
MLlaio2004, 5, 36, 56, 60
mom2par.gamma (P3), 61
moment_estimation, 5
moment_estimation (MLlaio2004), 56
moments, 4, 57
monthlyflows (hydroSIMN), 48
MSClaio2008, 5, 58

nearest (traceWminim), 71
nls, 3
Normal, 23
normplot (DISTPLOTS), 21
normpoints (DISTPLOTS), 21
nsRFA, 9, 14, 18, 20, 23, 26–28, 30, 32, 34, 36, 38, 40, 45, 47, 49, 51, 53, 56–58, 60, 64, 66, 68–70, 72, 74
nsRFA (nsRFA-package), 3
nsRFA-package, 3

optim, 51
P3, 26, 30, 32, 34, 40, 51, 56, 61, 71
par.exp (EXP), 24
par.gamma, 5
par.gamma (P3), 61
par.genlogis (GENLOGIS), 28
par.genpar, 5
par.genpar (GENPAR), 30
par.GEV, 5
par.GEV (GEV), 33
par.gumb (GUMBEL), 39
par.kappa (KAPPA), 50
par.lognorm (LOGNORM), 54
par2mom.gamma (P3), 61
parameters (hydroSIMN), 48
paretoplot (DISTPLOTS), 21
PEL (HW.original), 45
plot, 69
plot.BayesianMCMC (BayesianMCMC), 10
plot.BayesianMCMCreg (BayesianMCMC), 10
plot.HWorig (HW.original), 45
plot.MSClaio2008 (MSClaio2008), 58
plotBayesianMCMCreg_surf (BayesianMCMC), 10
plotpos (DISTPLOTS), 21
plotposRP (DISTPLOTS), 21
plotposRhist (DISTPLOTS), 21
pointspos (DISTPLOTS), 21
pointsposRP (DISTPLOTS), 21
pointsposRhist (DISTPLOTS), 21
predict.lm, 20, 66
predinterval.lm (REGRDIAGNOSTICS), 64
print.BayesianMCMC (BayesianMCMC), 10
print.BayesianMCMCreg (BayesianMCMC), 10
print.bestlm (bestlm), 17
print.HWorig (HW.original), 45
print.MSClaio2008 (MSClaio2008), 58
prt.lm (REGRDIAGNOSTICS), 64
qqnorm, 23
R2 (DIAGNOSTICS), 19
R2.lm (REGRDIAGNOSTICS), 64
rand.exp (EXP), 24
rand.gamma (P3), 61
rand.genlogis (GENLOGIS), 28
rand.genpar (GENPAR), 30
rand.GEV (GEV), 33
rand.gumb (GUMBEL), 39
rand.kappa (KAPPA), 50
rand.lognorm (LOGNORM), 54
regionalexpplot (DISTPLOTS), 21
regionalfrechetplot (DISTPLOTS), 21
regionalgumbelplot (DISTPLOTS), 21
regionallmoments (Lmoments), 52
regionallognormplot (DISTPLOTS), 21
regionalnormplot (DISTPLOTS), 21
regionalparetoplot (DISTPLOTS), 21
regionalplotpos (DISTPLOTS), 21
REGLMR (HW.original), 45
REGRDIAGNOSTICS, 4, 18, 20, 64
REGTST (HW.original), 45
RMSE (DIAGNOSTICS), 19
RMSE.lm (REGRDIAGNOSTICS), 64
RMSEjklm (REGRDIAGNOSTICS), 64
RMSEP (DIAGNOSTICS), 19
rnorm, 26, 30, 32, 34, 40, 51, 56, 64
roi, 4, 9, 38, 45, 67, 72
runif, 26, 30, 32, 34, 40, 51, 56, 64

SaintLaurent_cont (Ardechedata), 9
SaintMartin_cont (Ardechedata), 9
SaintMartin_hist (Ardechedata), 9
SAMLMR (HW.original), 45

SaintMartin_hist (Ardechedata), 9
SAMLMR (HW.original), 45
SAMLMU (HW.original), 45
SAMPWM (HW.original), 45
sd, 53, 58
serieplot (SERIESPLOTS), 69
SERIESPLOTS, 69
skew (moments), 57
STATICPLOTS, 70
stats, 3, 4
studentplot (DISTPLOTS), 21
studentpoints (DISTPLOTS), 21
summary.bestlm (bestlm), 17
summary.lm, 20, 66
summary.MSClaio2008 (MSClaio2008), 58
sumtraceW (traceWminim), 71

traceWminim, 4, 9, 38, 45, 68, 71
unifplot (DISTPLOTS), 21
unifpoints (DISTPLOTS), 21

var, 53, 58, 74
varLCA (varLmoments), 73
varLCV (varLmoments), 73
varLkur (varLmoments), 73
varLmoments, 5, 73
vif.lm (REGRDIAGNOSTICS), 64
Vogue_cont (Ardechedata), 9

W2 (GOFlaio2004), 35
weibullplot (DISTPLOTS), 21
weibullpoints (DISTPLOTS), 21