Package 'circularEV'

January 21, 2021

Type Package

Title Extreme Value Analysis for Circular Data

Version 0.1.0

Date 2021-01-19

Depends R (>= 3.6)

Imports parallel, foreach, doParallel, mgcv, circular, NPCirc, ggplot2, utils, stats

Author Evandro Konzen

Maintainer Evandro Konzen <e.konzen@reading.ac.uk>

Description General functions for performing extreme value analysis on a circular domain as part of the statistical methodology in the paper by Konzen, E., Neves, C., and Jonathan, P. (2020+). Modelling non-stationary extremes of storm severity: comparing parametric and semi-parametric inference. Environmetrics (to appear).

Acknowledgements This work was supported by the EPSRC UKRI Innovation Fellowship (grant number EP/S001263/1)

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.0.2

Suggests plotly, testthat, knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2021-01-21 11:10:02 UTC

R topics documented:

`CalcRLsplineML` *Calculate T-year levels for spline ML model*

Description

Calculate T-year levels for spline ML model

Usage

```
CalcRLsplineML(
  Data,
  drc,
  h,
  xiBoot,
  sigBoot,
  TTs = c(100, 10000),
  thetaGrid = 1:360,
  timeRange,
  thr
)
```

Arguments

`Data`	Response variable
`drc`	Directional covariate
`h`	Bandwidth value
`xiBoot`	Bootstrap estimates for EVI
`sigBoot`	Bootstrap estimates for shape
`TTs`	T-year levels. For example, TTs = c(100, 10000).
`thetaGrid`	Grid values at which the estimation is performed
`timeRange`	Time range of the sample
`thr`	Threshold values along thetaGrid

Value

List including bootstrap estimates of T-year levels.

See Also

SplineML for examples.

Examples

```
## See also examples in vignettes:
# vignette("localMethods", package = "circularEV")
# vignette("splineML", package = "circularEV")
```

---

drc *Directional covariate for HsSP data*

Description

A vector of length 1521.

Usage

```
drc
```

Format

A vector of length 1521.

Details

Directional covariate of HsSP data used by Reistad, M., Breivik, Ø., Haakenstad, H., Aarnes, O. J., Furevik, B. R., and Bidlot, J.-R. (2011), A high-resolution hindcast of wind and waves for the North Sea, the Norwegian Sea, and the Barents Sea, J. Geophys. Res., 116:1-18.

HsSP *Hindcast storm peak significant wave height data*

Description

A vector of length 1521.

Usage

```
HsSP
```

Format

A vector of length 1521.

Details

HsSP data used by Reistad, M., Breivik, Ø., Haakenstad, H., Aarnes, O. J., Furevik, B. R., and Bidlot, J.-R. (2011), A high-resolution hindcast of wind and waves for the North Sea, the Norwegian Sea, and the Barents Sea, J. Geophys. Res., 116:1-18.

`LocalEstim` *Local bootstrap estimation of EVI, scale and T-year levels*

Description

Local bootstrap estimation of EVI, scale and T-year levels

Usage

```
LocalEstim(
  Data,
  drc,
  thr = NULL,
  thetaGrid,
  nBoot = 100,
  EVIestimator = "Mom",
  h = 30,
  useKernel = TRUE,
  concent = 10,
  movThr = TRUE,
  TTs = NULL,
  timeRange = NULL
)
```

Arguments

`Data`	Response variable
`drc`	Directional covariate
`thr`	Threshold values along thetaGrid
`thetaGrid`	Grid values at which the estimation is performed
`nBoot`	Number of bootstrap resamples. Default to 100.
`EVIestimator`	It can be either "ML" or "Mom"
`h`	Bandwidth value
`useKernel`	Logical. If TRUE (default), use kernel to assign weights depending on the directional distance.

concent	Concentration parameter value for von Mises kernel
movThr	Logical. If TRUE (default), moving threshold within the window used.
TTs	T-year levels. For example, TTs = c(100, 10000).
timeRange	Time range of the sample

Details

See Konzen, E., Neves, C., and Jonathan, P. (2020+). Modelling non-stationary extremes of storm severity: comparing parametric and semi-parametric inference. Environmetrics (to appear).

Value

List including bootstrap estimates of EVI, scale and T-year levels.

Examples

```
data(HsSP)
data(drc)
timeRange <- 54.5

idx <- order(drc)
drc <- drc[idx]
Data <- HsSP[idx]
set.seed(1234)
Data <- Data + runif(length(Data), -1e-4, 1e-4)

thetaVec <- 1:360

data(thresholdExampleMom) # loads threshold example
thrResultMom <- thresholdExampleMom



h <- 60
useKernel <- TRUE
concent <- 10
movThr <- TRUE
nBoot <- 30
set.seed(1234)
output <- LocalEstim(Data=Data, drc=drc, thr=thrResultMom,
                     thetaGrid=thetaVec, nBoot=nBoot, EVIestimator="Mom", h=h,
                     useKernel=useKernel, concent=concent, movThr=movThr,
                     TTs=c(100, 10000), timeRange=timeRange)

RLBoot <- output$RLBoot

PlotParamEstim(bootEstimates=output$xiBoot, thetaGrid=thetaVec, ylab=bquote(hat(xi)),
               alpha=0.05, ylim=NULL, cex.axis=15, cex.lab=2, thrWidth=2)

PlotParamEstim(bootEstimates=output$sigBoot, thetaGrid=thetaVec, ylab=bquote(hat(sigma)),
               alpha=0.05, ylim=NULL, cex.axis=15, cex.lab=2, thrWidth=2)
```

```
# 100-year level
PlotRL(RLBootList=RLBoot, thetaGrid=thetaVec, Data=Data, drc=drc,
       TTs=c(100, 10000), whichPlot=1, alpha=0.05, ylim=NULL,
       pointSize=1, cex.axis=15, cex.lab=2, thrWidth=2)

PolarPlotRL(RLBootList=RLBoot, thetaGrid=thetaVec, Data=Data, drc=drc,
            TTs=c(100, 10000), whichPlot=1, alpha=0.05, ylim=NULL,
            pointSize=4, fontSize=12, lineWidth=2)



## See examples in vignette:
# vignette("localMethods", package = "circularEV")
```

`PlotData` *Plot of circular data*

Description

Plot of circular data

Usage

```
PlotData(
  Data,
  drc,
  thr = NULL,
  ylim = NULL,
  pointSize = 4,
  cex.axis = 15,
  cex.lab = 2,
  thrWidth = 2,
  thrColor = "#D45E1A",
  thrLineType = 1,
  ylab = NULL
)
```

Arguments

`Data`	Response variable
`drc`	Directional covariate
`thr`	Threshold values along thetaGrid
`ylim`	Range of values
`pointSize`	Size of points (observations)
`cex.axis`	Graphical parameter

cex.lab	Graphical parameter
thrWidth	Threshold width
thrColor	Threshold colour
thrLineType	Graphical parameter
ylab	y-axis label

Value

Plot of circular data, possibly including a threshold.

Examples

```
data(HsSP)
data(drc)

PlotData(Data=HsSP, drc=drc, thr=NULL, pointSize=1, cex.axis=15, cex.lab=2,
         thrWidth=2)

data(thresholdExampleML) # loads threshold example

PlotData(Data=HsSP, drc=drc, thr=thresholdExampleML, pointSize=1, cex.axis=15,
         cex.lab=2, thrWidth=2)
```

PlotParamEstim *Plot of parameter estimates with bootstrap confidence intervals*

Description

Plot of parameter estimates with bootstrap confidence intervals

Usage

```
PlotParamEstim(
  bootEstimates,
  thetaGrid = 1:360,
  alpha = 0.05,
  ylim = NULL,
  cex.axis = 15,
  cex.lab = 2,
  thrWidth = 2,
  ylab = NULL,
  thrColor = "#D45E1A"
)
```

Arguments

bootEstimates	Bootstrap estimates (for example, shape or scale)
thetaGrid	Grid values at which the estimation is performed
alpha	Significance level for the confidence intervals. Default to 0.05.
ylim	Range for the y-axis
cex.axis	Graphical parameter
cex.lab	Graphical parameter
thrWidth	Threshold width
ylab	y-axis label
thrColor	Threshold colour

Value

Plot of parameter estimates.

See Also

SplineML and LocalEstim for examples.

Examples

```
## See examples in vignettes:
# vignette("localMethods", package = "circularEV")
# vignette("splineML", package = "circularEV")
```

PlotRL *Plot of T-year levels*

Description

Plot of T-year levels

Usage

```
PlotRL(
  RLBootList,
  Data,
  drc,
  thetaGrid = 1:360,
  TTs,
  whichPlot,
  alpha = 0.05,
  ylim = NULL,
  pointSize = 1,
  cex.axis = 15,
```

```
  cex.lab = 2,
  thrWidth = 2,
  thrColor = "#D45E1A",
  ylab = NULL
)
```

Arguments

RLBootList	List containing bootstrap estimates of T-year levels
Data	Response variable
drc	Directional covariate
thetaGrid	Grid values at which the estimation is performed
TTs	T-year levels. For example, TTs = c(100, 10000).
whichPlot	Index identifying which T-year level should be plotted from TTs. If TTs = c(100, 10000), then whichPlot=2 produces a plot for the 10000-year level
alpha	Significance level for the confidence intervals. Default to 0.05.
ylim	Range for the y-axis
pointSize	Size of points (observations)
cex.axis	Graphical parameter
cex.lab	Graphical parameter
thrWidth	Threshold width
thrColor	Threshold colour
ylab	y-axis label

Value

Plot of T-year levels.

See Also

SplineML and LocalEstim for examples.

Examples

```
## See also examples in vignettes:
# vignette("localMethods", package = "circularEV")
# vignette("splineML", package = "circularEV")
```

PolarPlotData *Polar plot of circular data*

Description

Polar plot of circular data

Usage

```
PolarPlotData(
  Data,
  drc,
  thr = NULL,
  ylim = NULL,
  pointSize = 1,
  fontSize = 12,
  thrWidth = 4,
  thrColor = "#D45E1A"
)
```

Arguments

Data	Response variable
drc	Directional covariate
thr	Threshold values along thetaGrid
ylim	Range of values
pointSize	Size of points (observations)
fontSize	Font size
thrWidth	Threshold width
thrColor	Threshold colour

Value

Polar plot of circular data, possibly including a threshold

Examples

```
data(HsSP)
data(drc)

PolarPlotData(Data=HsSP, drc=drc, thr=NULL, pointSize=4, fontSize=14,
              thrWidth=4, ylim=c(0,max(HsSP)))

data(thresholdExampleML) # loads threshold example

PolarPlotData(Data=HsSP, drc=drc, thr=thresholdExampleML, pointSize=4,
              fontSize=12, thrWidth=4, ylim=c(0,max(HsSP)))
```

PolarPlotRL *Polar plot of T-year levels*

Description

Polar plot of T-year levels

Usage

```
PolarPlotRL(
  RLBootList,
  Data,
  drc,
  thetaGrid = 1:360,
  TTs,
  whichPlot,
  alpha = 0.05,
  ylim = NULL,
  pointSize = 4,
  fontSize = 12,
  lineWidth = 4
)
```

Arguments

RLBootList	List containing bootstrap estimates of T-year levels
Data	Response variable
drc	Directional covariate
thetaGrid	Grid values at which the estimation is performed
TTs	T-year levels. For example, TTs = c(100, 10000).
whichPlot	Index identifying which T-year level should be plotted from TTs. If TTs = c(100, 10000), then whichPlot=2 produces a plot for the 10000-year level
alpha	Significance level for the confidence intervals. Default to 0.05.
ylim	Range for the y-axis
pointSize	Size of points (observations)
fontSize	Font size
lineWidth	Threshold width

Value

Polar plot of T-year levels.

See Also

SplineML and LocalEstim for examples.

Examples

```
## See also examples in vignettes:
# vignette("localMethods", package = "circularEV")
# vignette("splineML", package = "circularEV")
```

SplineML *Spline ML fitting*

Description

Spline ML fitting

Usage

```
SplineML(
  excesses,
  drc,
  thetaVec = 0:360,
  nBoot = 100,
  numIntKnots = 10,
  knotsType = "eqSpaced",
  lambda = seq(0, 2, by = 0.5),
  kappa = seq(0, 2, by = 0.5),
  nCandidatesInit = 1000,
  numCores = 2
)
```

Arguments

excesses	Excesses data
drc	Directional covariate
thetaVec	Grid values at which the threshold will be evaluated
nBoot	Number of bootstrap resamples
numIntKnots	Number of internal knots
knotsType	Position of knots. Default to "eqSpaced". Otherwise, the knots will be placed at the quantiles of observed directions.
lambda	Penalty parameter values for lambda
kappa	Penalty parameter values for kappa
nCandidatesInit	Number of initial parameter vectors. Optimisation will start with the best.
numCores	Number of CPU cores to be used

Details

See Konzen, E., Neves, C., and Jonathan, P. (2020+). Modelling non-stationary extremes of storm severity: comparing parametric and semi-parametric inference. Environmetrics (to appear).

Value

List of bootstrap estimates of shape and scale, and optimal values of lambda and kappa.

Examples

```
data(HsSP)
data(drc)
timeRange <- 54.5

idx <- order(drc)
drc <- drc[idx]
Data <- HsSP[idx]
set.seed(1234)
Data <- Data + runif(length(Data), -1e-4, 1e-4)

thetaVec <- 1:360

data(thresholdExampleML) # loads threshold example
thrResultML <- thresholdExampleML

lambda <- 100
kappa <- 40

thrPerObs <- thrResultML[drc]
excess <- Data - thrPerObs
drcExcess <- drc[excess>0]
excess <- excess[excess>0]



splineFit <- SplineML(excesses = excess, drc = drcExcess, nBoot = 30,
                      numIntKnots = 16, lambda=lambda, kappa=kappa, numCores=2)

xiBoot <- splineFit$xi
sigBoot <- splineFit$sig

PlotParamEstim(bootEstimates=xiBoot, thetaGrid=0:360, ylab=bquote(hat(xi)),
               alpha=0.05, ylim=NULL, cex.axis=15, cex.lab=2, thrWidth=2)

PlotParamEstim(bootEstimates=sigBoot, thetaGrid=0:360, ylab=bquote(hat(sigma)),
               alpha=0.05, ylim=NULL, cex.axis=15, cex.lab=2, thrWidth=2)

h <- 60 # needed for calculating local probability of exceedances
RLBoot <- CalcRLsplineML(Data=Data, drc=drc, xiBoot=xiBoot, sigBoot=sigBoot, h=h,
                         TTs=c(100, 10000), thetaGrid=thetaVec,
                         timeRange=timeRange, thr=thrResultML)

# 100-year level
PlotRL(RLBootList=RLBoot, thetaGrid=thetaVec, Data=Data, drc=drc,
       TTs=c(100, 10000), whichPlot=1, alpha=0.05, ylim=NULL,
       pointSize=1, cex.axis=15, cex.lab=2, thrWidth=2)
```

```
PolarPlotRL(RLBootList=RLBoot, thetaGrid=thetaVec, Data=Data, drc=drc,
            TTs=c(100, 10000), whichPlot=1, alpha=0.05, ylim=c(0, 25),
            pointSize=4, fontSize=12, lineWidth=2)



## See also examples in vignette:
# vignette("splineML", package = "circularEV")
```

`thresholdExampleML` *Threshold for spline ML and local ML examples*

Description

A vector of threshold values at directions 1,...,360. It is used for spline ML and local ML examples.

Usage

```
thresholdExampleML
```

Format

A vector of 360 values.

Details

It has been generated as follows:

```
data(HsSP)
data(drc)
timeRange <- 54.5

idx <- order(drc)
drc <- drc[idx]
Data <- HsSP[idx]
set.seed(1234)
Data <- Data + runif(length(Data), -1e-4, 1e-4)

thetaVec <- 1:360

thresholdExampleML <- ThrSelection(Data=Data, drc=drc, h=60, b=0.35,
                                   thetaGrid=thetaVec, EVIestimator="ML",
                                useKernel=T, concent=10, bw=30, numCores=2)$thr
```

thresholdExampleMom *Threshold for local Moment examples*

Description

A vector of threshold values at directions 1,...,360. It is used for local Moment examples.

Usage

```
thresholdExampleMom
```

Format

A vector of 360 values.

Details

It has been generated as follows:

```
data(HsSP)
data(drc)
timeRange <- 54.5

idx <- order(drc)
drc <- drc[idx]
Data <- HsSP[idx]
set.seed(1234)
Data <- Data + runif(length(Data), -1e-4, 1e-4)

thetaVec <- 1:360

thresholdExampleMom <- ThrSelection(Data=Data, drc=drc, h=60, b=0.35,
                                    thetaGrid=thetaVec, EVIestimator="Mom",
                                useKernel=T, concent=10, bw=30, numCores=2)$thr
```

ThrSelection *Threshold selection*

Description

This function selects a moving threshold for circular data using an automatic procedure for selecting the local number of exceedances

Usage

```
ThrSelection(
  Data,
  drc,
  h = 30,
  b = 0.35,
  thetaGrid,
  EVIestimator = "ML",
  useKernel = TRUE,
  concent = 10,
  bw = 30,
  numCores = 2
)
```

Arguments

Data	Response variable
drc	Directional covariate
h	Bandwidth value
b	Parameter used in the automatic procedure for selection of local number of exceedances
thetaGrid	Grid values at which the estimation is performed
EVIestimator	It can be either "ML" or "Mom"
useKernel	Logical. If TRUE (default), use kernel to assign weights depending on the directional distance.
concent	Concentration parameter value for von Mises kernel
bw	Bandwidth parameter value for smoothing the sample path of the selected threshold
numCores	Number of CPU cores to be used

Details

See Konzen, E., Neves, C., and Jonathan, P. (2020+). Modelling non-stationary extremes of storm severity: comparing parametric and semi-parametric inference. Environmetrics (to appear).

Value

List containing the selected threshold and selected number of local exceedances at each direction in the grid.

See Also

PlotData and PolarPlotData to see how the threshold can be visualised.

Examples

```
data(HsSP)
data(drc)
timeRange <- 54.5

idx <- order(drc)
drc <- drc[idx]
Data <- HsSP[idx]
set.seed(1234)
Data <- Data + runif(length(Data), -1e-4, 1e-4)

thetaVec <- 1:360



thrResultMom <- ThrSelection(Data=Data, drc=drc, h=60, b=0.35, thetaGrid=thetaVec,
                             EVIestimator="Mom", useKernel=T, concent=10, bw=30,
                             numCores=2)$thr
thrResultML <- ThrSelection(Data=Data, drc=drc, h=60, b=0.35, thetaGrid=thetaVec,
                            EVIestimator="ML", useKernel=T, concent=10, bw=30,
                            numCores=2)$thr



## See also examples in vignettes:
# vignette("localMethods", package = "circularEV")
# vignette("splineML", package = "circularEV")
```

Index