Package ‘NPCirc’

January 20, 2025

Type Package

Title Nonparametric Circular Methods

Version 3.1.1

Date 2022-05-24

Author Maria Alonso-Pena [aut, cre],
Maria Oliveira [aut],
Jose Ameijeiras-Alonso [aut],
Rosa M. Crujeiras [aut],
Irene Gijbels [aut],
Alberto Rodriguez-Casal [aut],
Felicita Scapini [dtc]

Maintainer Maria Alonso-Pena <maria.alonsopena@kuleuven.be>

Description

Nonparametric smoothing methods for density and regression estimation involving circular data, including the estimation of the mean regression function and other conditional characteristics.

Depends R(>= 3.5.0), circular

License GPL (>= 2)

Imports Rcpp (>= 1.0.8.3), misc3d, movMF, plotrix, rgl, shape,
Bolstad2

LinkingTo Rcpp, RcppArmadillo

URL https://www.jstatsoft.org/v61/i09/

NeedsCompilation yes

Repository CRAN

Date/Publication 2022-11-10 13:00:11 UTC

Contents

`NPCirc-package` *Nonparametric circular methods.*

Description

Nonparametric smoothing methods for density and regression estimation involving circular data, including methods for estimating circular densities and mean regression functions, described in Oliveira et al. (2014), testing proposals for circular mean regression described in Alonso-Pena et al. (2021), estimation of conditional characteristics (Alonso-Pena et al. 2022a,2022b) and estimation of conditional modes (Alonso-Pena and Crujeiras, 2022).

Details

Package:	NPCirc
Type:	Package
Version:	4.0.1
Date:	2021-07-21
License:	GPL-2
LazyLoad:	yes

This package incorporates the function `kern.den.circ` which computes the circular kernel density estimator. For choosing the smoothing parameter different functions are available: `bw.rt`, `bw.CV`, `bw.pi`, and `bw.boot`. For mean regression involving circular variables, the package includes the functions: `kern.reg.circ.lin` for a circular covariate and linear response; `kern.reg.circ.circ` for a circular covariate and a circular response; `kern.reg.lin.circ` for a linear covariate and a circular response. The three functions compute Nadaraya-Watson and Local-Linear smoothers. The functions `bw.reg.circ.lin`, `bw.reg.circ.circ` and `bw.reg.circ.lin` implement cross–validation rules for selecting the smoothing parameter. Functions `circsizer.density` and `circsizer.regression` provide CircSiZer maps for kernel density estimation and regression estimation, respectively. Functions `noeffect.circ.lin`, `noeffect.circ.circ` and `noeffect.lin.circ` compute the test of no effect to assess the significance of the predictor variable. Additionally, functions `ancova.circ.lin`, `ancova.circ.circ` and `ancova.lin.circ` implement hypothesis testing tools to assess the equality and parallelism of regression curves across different groups of observations.

Function `circ.local.lik` implements the estimation of different functions of interest (transformations of the mean function) in contexts where the predictor variable is circular and the conditional distribution is a Gaussian, Bernoulli, Poisson or gamma. Function `bw.circ.local.lik` computes smoothing parameters for the estimation in the previously described cases, allowing for three different rules. Function `kern.dpreg.circ` implements the joint kernel estimation of the mean and dispersion functions in cases where the predictor is circular and the conditional distribution is a double Poisson, a particular case of the double exponential family. Smoothing parameters in this context can be computed with function `bw.joint.dpcirc`, using a two-step cross-validation algorithm.

Functions `modalreg.circ.circ`, `modalreg.lin.circ` and `modalreg.circ.lin` implement the estimation of the modal regression multifunction (the conditional local modes) in the three circular regression scenarios. Smoothing parameters for these contexts can by computed by modal cross-validation employin functions `bw.modalreg.circ.circ`, `bw.modalreg.lin.circ` and `bw.modalreg.circ.lin`.

Functions `dcircmix` and `rcircmix` compute the density function and generate random samples of a circular distribution or a mixture of circular distributions, allowing for different components such as the circular uniform, von Mises, cardioid, wrapped Cauchy, wrapped normal and wrapped skew-normal. Finally, some data sets are provided. Missing data are allowed. Registries with missing data are simply removed.

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

Acknowledgements

This work has been supported by Project MTM2008-03010 from the Spanish Ministry of Science; Project and MTM201676969-P from the AEI co-funded by the European Regional Development

Fund (ERDF), the Competitive Reference Groups 2017-2020 (ED431C 2017/38) from the Xunta de Galicia through the ERDF; and Innovation IAP network (Developing crucial Statistical methods for Understanding major complex Dynamic Systems in natural, biomedical and social sciences (StUDyS)) from Belgian Science Policy. Work of Maria Alonso-Pena was supported by grant ED481A-2019/139 from Xunta de Galicia. Work of Jose Ameijeiras-Alonso was supported by the FWO research project G.0826.15N (Flemish Science Foundation); and GOA/12/014 project (Research Fund KU Leuven). The authors want to acknowledge Prof. Arthur Pewsey for facilitating data examples and for his comments.

Author(s)

Maria Oliveira, Maria Alonso-Pena, Jose Ameijeiras-Alonso, Rosa M. Crujeiras, Alberto Rodriguez–Casal and Irene Gijbels.

Maintainer: Maria Alonso-Pena <mariaalonso.pena@usc.es>

References

Oliveira, M., Crujeiras, R.M. and Rodriguez–Casal, A. (2012) A plug–in rule for bandwidth selection in circular density. *Computational Statistics and Data Analysis*, **56**, 3898–3908.

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2013) Nonparametric circular methods for exploring environmental data. *Environmental and Ecological Statistics*, **20**, 1–17.

Oliveira, M., Crujeiras, R.M. and Rodriguez–Casal (2014) CircSiZer: an exploratory tool for circular data. *Environmental and Ecological Statistics*, **21**, 143–159.

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2014) NPCirc: an R package for nonparametric circular methods. *Journal of Statistical Software*, **61**(9), 1–26. https://www.jstatsoft.org/v61/i09/

Alonso-Pena, M., Ameijeiras-Alonso, J. and Crujeiras, R.M. (2021) Nonparametric tests for circular regression. *Journal of Statistical Computation and Simulation*, **91**(3), 477–500.

Alonso-Pena, M., Gijbels, I. and Crujeiras, R.M. (2022a). A general framework for circular local likelihood regression. *Under review*.

Alonso-Pena, M., Gijbels, I. and Crujeiras, R.M. (2022b). Flexible joint modeling of mean and dispersion for the directional tuning of neuronal spike counts. *Under review*.

Alonso-Pena, M. and Crujeiras, R. M. (2022). Analizing animal escape data with circular nonparametric multimodal regression. Annals of Applied Statistics. (To appear).

`ancova.circ.lin` *Nonparametric analysis of covariance tests for circular regression*

Description

Function `ancova.circ.lin` computes nonparametric ANCOVA tests to compare regression curves with a circular predictor variable and a real-valued response variable. The null hypothesis may be either equality or parallelism of the regression curves, as described in Alonso-Pena et al. (2021). It uses the nonparametric Nadaraya-Watson estimator or the Local-Linear estimator for circular-linear data described in Di Marzio et al. (2009) and Oliveira et al. (2013).

Function ancova.lin.circ computes nonparametric ANCOVA tests to compare regression curves with a real-valued predictor variable and a circular response variable. The null hypothesis may be either equality or parallelism of the regression curves, as described in Alonso-Pena et al. (2021). It uses the nonparametric Nadaraya-Watson estimator or the Local-Linear estimator for linear-circular data described in Di Marzio et al. (2012).

Function ancova.circ.circ computes nonparametric ANCOVA tests to compare regression curves with a circular predictor variable and a circular response variable. The null hypothesis may be either equality or parallelism of the regression curves, as described in Alonso-Pena et al. (2021). It uses the nonparametric Nadaraya-Watson estimator or the Local-Linear estimator for circular-circular data described in Di Marzio et al. (2012).

Usage

```
ancova.circ.lin(x, y, g, bw, bw1, test = "eq", method = "LL",calib = "chisq",
   n_boot = 500)
ancova.lin.circ(x, y, g, bw, bw1, test = "eq", method = "LL", n_boot = 500)
ancova.circ.circ(x, y, g, bw, bw1, test = "eq", method = "LL", n_boot = 500)
```

Arguments

x	Vector of data for the independent variable. The object is coerced to class circular when using functions ancova.circ.lin and ancova.circ.circ.
y	Vector of data for the dependent variable. This must be same length as x. The object is coerced to class circular when using functions ancova.lin.circ and ancova.circ.circ.
g	Vector of group indicators.
bw	Smoothing parameter to be used. If not provided it selects the parameter obtained by cross-validation.
bw1	Preliminary smoothing parameter for the parallelism test.
test	Character string giving the type of test to be performed. Must be one of "eq" for the test of equality or "paral" for the test of parallelism. Default is test="eq".
method	Character string giving the estimator to be used. This must be one of "LL" for Local-Linear estimator or "NW" for Nadaraya-Watson estimator. Default method="LL".
calib	Character string giving the calibration method to be used in ancova.circ.lin function. This must be one of "chisq" for the chi-squared approximation or "boot" for the bootstrap calibration.
n_boot	Number of bootstrap resamples. Default is n_boot=500. In function ancova.circ.lin, only if calib="boot".

Details

See Alonso-Pena et al. (2021). The NAs will be automatically removed.

Value

A list with class "htest" containing the following components:

statistic	observed value of the statistic.
bw	Smoothing parameter used.
p.value	p-value for the test.
data.name	a character string giving the name(s) of the data.
alternative	a character string describing the alternative hypothesis.

Author(s)

Maria Alonso-Pena, Jose Ameijeiras-Alonso and Rosa M. Crujeiras

References

Alonso-Pena, M., Ameijeiras-Alonso, J. and Crujeiras, R.M. (2021) Nonparametric tests for circular regression. *Journal of Statistical Computation and Simulation*, **91**(3), 477–500.

Di Marzio, M., Panzera A. and Taylor, C. C. (2009) Local polynomial regression for circular predictors. *Statistics and Probability Letters*, **79**, 2066–2075.

Di Marzio, M., Panzera A. and Taylor, C. C. (2012) Non–parametric regression for circular responses. *Scandinavian Journal of Statistics*, **40**, 228–255.

Oliveira, M., Crujeiras R.M. and Rodriguez-Casal, A. (2013) Nonparametric circular methods for exploring environmental data. *Environmental and Ecological Statistics*, **20**, 1–17.

See Also

kern.reg.circ.lin, kern.reg.lin.circ, kern.reg.circ.circ

Examples

```
# ANCOVA circ-lin
set.seed(2025)
x1 <- rcircularuniform(100)
x2 <- rcircularuniform(100)
x <- c(x1, x2)
y1 <- 2*sin(as.numeric(x1)) + rnorm(100, sd=2)
y2 <- 4 + 2*sin(as.numeric(x2)) + rnorm(100, sd=2)
y <- c(y1, y2)
g <- c(rep(0,100), rep(1,100))
ancova.circ.lin(x, y, g,  test = "eq")
ancova.circ.lin(x, y, g,  test = "paral")


# ANCOVA lin-circ
set.seed(2025)
x1 <- runif(100)
x2 <- runif(100)
y1 <- 3*pi*x1^2 + rvonmises(100, mu = 0, kappa = 6)
```

```
y2 <- 2*pi/8 + 3*pi*x2^2 + rvonmises(100, mu = 0, kappa = 6)
x <- c(x1, x2)
y <- c(y1, y2)
g<-c(rep(0, 100), rep(1, 100))
ancova.lin.circ(x, y, g,  test = "eq")
ancova.lin.circ(x, y, g,  test = "paral")


# ANCOVA circ-circ
set.seed(2025)
x1 <- rcircularuniform(100)
x2 <- rcircularuniform(100)
y1 <- 2*sin(2*x1) + rvonmises(100, mu = 0, kappa = 8 )
y2 <- pi/8 + 2*sin(2*x2) + rvonmises(100, mu = 0, kappa = 8 )
x <- c(x1, x2)
y <- c(y1, y2)
g<-c(rep(0, 100), rep(1, 100))
ancova.circ.circ(x, y, g,  test = "eq")
ancova.circ.circ(x, y, g,  test = "paral")
```

`bw.AA` *Direct and solve-the-equation plug-in rule*

Description

Smoothing selectors for the circular kernel density (and its derivatives) estimator. This function implements the l-stage solve-the-equation and direct plug-in smoothing selector.

Usage

```
bw.AA(x,deriv.order=0,method = c("ste","dpi"),nstage=2,kernel="vonmises",M=NULL,
      commonkappa=TRUE,Q1=NULL,Q2=NULL,lower=NULL,upper=NULL,tol=NULL,
      approximate=NULL)
```

Arguments

`x`	Data from which the smoothing parameter is to be computed. The object is coerced to class `circular`.
`deriv.order`	Derivative order. Default `deriv.order=0` (density estimation).
`method`	Either `"ste"` (solve-the-equation) or `"dpi"` (direct plug-in). Can be abbreviated.
`nstage`	Number of stages in the plug-in smoothing parameter. Default `nstage=2`.
`kernel`	A character string giving the smoothing kernel to be used. This must be one of `vonmises` or `wrappednormal`. Default `kernel=vonmises`.
`M`	Integer indicating the number of components in the von Mises mixture at stage 0. If `M` is a vector, AIC will be used, to select the number of components between the indicated values of `M`. Default, `M=1` if method `"ste"`, `M=1:5` if method `"dpi"`.

commonkappa
: Logical; if TRUE, at stage 0, all the components in the von Mises mixture have the same concentration. Default, commonkappa=T.

Q1
: Vector of constants related to the kernel to derive the explicit expression of the optimal smoothing parameters of the density functionals. Its value is provided by default when using the "vonmises" or "wrappednormal" as kernel.

Q2
: Constant related to the kernel to derive the explicit expression of the optimal smoothing parameters of the density derivative. Its value is provided by default when using the "vonmises" or "wrappednormal" as kernel.

lower, upper
: For method "ste", the range over which the bandwidth h is searched. Default, lower=10^(-3) and upper=pi^2/3 if kernel="vonmises"; lower=10^(-10) and upper=0.5, otherwise.

tol
: For method "ste", the convergence tolerance for searching the smoothing parameter with uniroot. Default, tol=.Machine$double.eps^0.25.

approximate
: For method "dpi", logical, if TRUE, the explicit expressions (relying on asymptotics) for the optimal smoothing parameters are employed. If FALSE, an optimization routine is employed, searching for the smoothing parameter minimizing the asymptotic mean squared error of the density derivatives and functionals. Default, approximate=T.

Details

By default, this function computes the solve-the-equation plug-in rule for circular kernel density estimation. If method="dpi", this function computes the direct plug-in rule.

At stage 0, a mixture of von Mises is employed for computing the rule of thumb. The reason for employing a mixture model is that the von Mises estimates a uniform density when the true density model is k-fold rotational symmetric. Thus, in that case, the density functional estimator would be close to zero.

The number of components in the mixture is controlled with M. By default, a von Mises density (M=1) is employed in method="ste". For method="dpi", by default, the number of components in the mixture is selected using the Akaike Information Criterion, by comparing a mixture of 1 to 5 components. For simplicity, by default, the same concentration parameter is employed in all the components. This may be changed by setting commonkappa=F.

For method="ste", the minimum number of stages is two (nstage=2). Otherwise, the solve-the-equation rule cannot be computed. A rule of thumb can be computed with method="dpi" and nstage=0.

Value

Value of the smoothing parameter (mean resultant length). When the kernel is vonmises the bandwidth is equal to the concentration parameter.

Author(s)

Jose Ameijeiras-Alonso.

References

Ameijeiras-Alonso, J. (2022). A reliable data-based smoothing parameter selection method for circular kernel estimation.

See Also

`density.circular`, kern.den.circ, `bw.pi`, `bw.rt`, `bw.CV`, `bw.boot`

Examples

```
set.seed(2022)
n <- 50
x <- rcircmix(n,model=6)
# Concentration parameter for density estimation
bw.AA(x) # Solve-the-equation concentration parameter
bw.AA(x, method="dpi") # Direct plug-in concentration parameter
# Concentration parameter for the density derivative estimate
bw.AA(x, method="ste") # Solve-the-equation concentration parameter
bw.AA(x, deriv.order=1, method="dpi") # Direct plug-in concentration parameter
```

`bw.boot` *Bootstrap method*

Description

This function implements the bootstrap procedure proposed by Di Marzio et al. (2011) for selecting the smoothing parameter for density estimation taking the von Mises density as kernel.

Usage

```
bw.boot(x, lower=0, upper=100, np=500, tol=0.1)
```

Arguments

`x`	Data from which the smoothing parameter is to be computed. The object is coerced to class `circular`.
`lower, upper`	`lower` and `upper` boundary of the interval to be used in the search for the value of the smoothing parameter. Default `lower=0` and `upper=100`.
`np`	Number of points where to evaluate the estimator for numerical integration. Default `np=500`.
`tol`	Convergence tolerance for `optimize`.

Details

This method is based on the proposal of Taylor (1989) for linear data. See also Oliveira et al. (2012). The NAs will be automatically removed.

Value

Value of the smoothing parameter.

Author(s)

Maria Oliveira, Rosa M. Crujeiras and Alberto Rodriguez–Casal

References

Di Marzio, M., Panzera A. and Taylor, C.C. (2011) Kernel density estimation on the torus. *Journal of Statistical Planning and Inference*, **141**, 2156–2173.

Oliveira, M., Crujeiras, R.M. and Rodriguez–Casal, A. (2012) A plug–in rule for bandwidth selection in circular density. *Computational Statistics and Data Analysis*, **56**, 3898–3908.

Taylor, C.C. (1989) Bootstrap choice of the smoothing parameter in kernel density estimation. *Biometrika*, **76**, 705–712.

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2014) NPCirc: an R package for nonparametric circular methods. *Journal of Statistical Software*, **61**(9), 1–26. https://www.jstatsoft.org/v61/i09/

See Also

kern.den.circ, bw.rt, bw.CV, bw.pi

Examples

```
set.seed(2012)
n <- 100
x <- rcircmix(n, model=17)
bw.boot(x, lower=0, upper=20)
```

---

bw.circ.local.lik *Smoothing parameter selection for circular local likelihood regression*

Description

Function bw.circ.local.lik computes values of the smoothing (concentration) parameter for local likelihood regression when the predictor is circular and the conditional density is either gaussian, Bernoulli, Poisson or gamma. The smoothing parameter can be selected by the refined rule, the CRSC rule or cross-validation, as described in Alonso-Pena et al. (2022).

Usage

```
bw.circ.local.lik(x, y, t, rule = NULL, p, family, startv = NULL,
  lower = 0, upper = 50, lower_ast = 0, upper_ast = 15, tol = 0.00001,
  maxit = 300, from = circular(0), to = circular(2 * pi), len = 250)
```

Arguments

- `x` — Vector of data for the independent variable. The object is coerced to class `circular`.
- `y` — Vector of data for the dependent variable. This must be same length as `x`.
- `t` — Points where the regression function is estimated. If `NULL`, equally spaced points are used according to the parameters `from`, `to` and `len`.
- `rule` — Character string giving the rule to be used to select the smoothing (concentration) parameter. This must be one of `"refined"` (only for `p = 1`), `"CRSC"` or `"cv"`.
- `p` — Degree of the local sine-polynomial to be used in the estimation process. It must be 1 or 3.
- `family` — Character string indicating the conditional density to be used. It must be one of `"gaussian"`, `"bernoulli"`, `"poisson"` or `"gamma"`.
- `startv` — Vector containing the initial values for the estimation algorithm if family is set as `"bernoulli"`, `"poisson"` or `"gamma"`. The vector must be of length 2 if `p = 1` and length 4 if `p = 3`. If `NULL`, the initial parameters are the ones corresponding to the global mean of the responses.
- `lower, upper` — Lower and upper boundary of the interval to be used in the search for the value of the smoothing parameter. Default `lower = 0` and `upper = 50`.
- `lower_ast, upper_ast` — Lower and upper boundary of the interval to be used in the search for the value of the pilot smoothing parameter in the refined rule. Default `lower_ast = 0` and `upper_ast = 15`.
- `tol` — Tolerance parameter for convergence in the numerical estimation. Only needed if family is one of `"bernoulli"`, `"poisson"` or `"gamma"`. Default is `tol = 0.00001`.
- `maxit` — Maximum number of iterations in the numerical estimation. Only needed if family is one of `"bernoulli"`, `"poisson"` or `"gamma"`. Default is `maxit = 300`.
- `from, to` — Left and right-most points of the grid at which the density is to be estimated. The objects are coerced to class `circular`.
- `len` — Number of equally spaced points at which the density is to be estimated.

Details

For the `refined` rule, which is only available for `p = 1`, first a pilot concentration parameter is selected with the (E)CRSC rule using a sine-polynomial of degree 3, where the search is conducted between the values `lower_ast` and `upper_ast`. With the pilot smoothing parameter, the MISE of the estimator is approximated and the refined rule selects the parameter which minimizes the approximated MISE.

The `CRSC` rule selects the parameter minimizing the Circular Residual Squares Criterion if `family = "gaussian"` and the Extended Circular Residual Squares Criterion in the other cases.

The `cv` rule performs a cross-validation search.

See Alonso-Pena et al. (2022) for more details.

Value

Value of the smoothing parameter.

Author(s)

Maria Alonso-Pena, Irene Gijbels and Rosa M. Crujeiras.

References

Alonso-Pena, M., Gijbels, I. and Crujeiras, R.M. (2022). A general framework for circular local likelihood regression. *Under review*.

See Also

circ.local.lik

Examples

```
data(spikes)
direction<-circular(spikes$direction,units="degrees")
counts<-spikes$counts
bw.circ.local.lik(direction, counts, rule = "refined", p=1, family="poisson")
```

---

bw.CV *Cross-validation for density estimation*

Description

This function provides a least squares cross-validation smoothing parameter or a likelihood cross-validation smoothing parameter for density estimation.

Usage

```
bw.CV(x, method="LCV", lower=0, upper=50, tol=1e-2, np=500)
```

Arguments

`x`	Data from which the smoothing parameter is to be computed. The object is coerced to class `circular`.
`method`	Character string giving the cross-validation rule to be used. This must be one of `"LCV"` or `"LSCV"`. Default `method="LCV"`.
`lower, upper`	`lower` and `upper` boundary of the interval to be used in the search for the value of the smoothing parameter. Default `lower=0` and `upper=50`.
`tol`	Convergence tolerance for `optimize`. Default `tol=1e-2`.
`np`	Number of points where to evaluate the estimator for numerical integration when `method="LSCV"`. Default `np=500`.

Details

The LCV smoothing parameter is obtained as the value of ν that maximizes the logarithm of the likelihood cross-validation function (8) in Oliveira et al. (2013). The LSCV smoothing parameter is obtained as the value of ν that minimizes expression (7) in Oliveira et al. (2013). See also Hall et al. (1987) and Oliveira et al. (2012). The NAs will be automatically removed.

Value

Value of the smoothing parameter.

Author(s)

Maria Oliveira, Rosa M. Crujeiras and Alberto Rodriguez–Casal

References

Hall, P., Watson, G.S. and Cabrera, J. (1987) Kernel density estimation with spherical data, *Biometrika*, **74**, 751–762.

Oliveira, M., Crujeiras, R.M. and Rodriguez–Casal, A. (2012) A plug–in rule for bandwidth selection in circular density. *Computational Statistics and Data Analysis*, **56**, 3898–3908.

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2013) Nonparametric circular methods for exploring environmental data. *Environmental and Ecological Statistics*, **20**, 1–17.

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2014) NPCirc: an R package for nonparametric circular methods. *Journal of Statistical Software*, **61**(9), 1–26. https://www.jstatsoft.org/v61/i09/

See Also

kern.den.circ, bw.rt, bw.pi, bw.boot

Examples

```
set.seed(2012)
n <- 100
x <- rcircmix(n, model=11)
bw.CV(x, method="LCV", lower=0, upper=20)
bw.CV(x, method="LSCV", lower=0, upper=20)
```

---

bw.joint.dpcirc *Smoothing parameter selection for circular double Poisson regression*

Description

Function bw.joint.dpcirc provides the smoothing parameters for the nonparametric joint estimator of the mean and dispersion functions when the conditional density is a double Poisson. It performs a joint cross-validation search.

Usage

```
bw.joint.dpcirc(x, y, startvmu = NULL, startvgam = NULL, lower=c(0.05,0.05),
  upper=c(50,7),tol = 0.00001, maxit = 300)
```

Arguments

`x`	Vector of data for the independent variable. The object is coerced to class circular.
`y`	Vector of data for the dependent variable. This must be same length as x and should contain counts.
`startvmu`	Vector of length two containing the initial values for the parameters corresponding to the estimation of the mean.
`startvgam`	Vector of length two containing the initial values for the parameters corresponding to the estimation of the dispersion.
`lower, upper`	Vectors of length two with the `lower` and `upper` boundaries of the intervals to be used in the search for the values of the smoothing parameters. The first component of each corresponds to the parameter associated to the estimation of the mean, while the second component corresponds to the estimation of the dispersion.
`tol`	Tolerance parameter for convergence in the numerical estimation.
`maxit`	Maximum number of iterations in the numerical estimation.

Details

See Alonso-Pena et al. (2022) for details.

The NAs will be automatically removed.

Value

A vector of length two with the first component being the value of the smoothing parameter associated to the mean estimation and with the second component being the value of the smoothing parameter associated to the dispersion estimation.

Author(s)

Maria Alonso-Pena, Irene Gijbels and Rosa M. Crujeiras

References

Alonso-Pena, M., Gijbels, I. and Crujeiras, R.M. (2022). Flexible joint modeling of mean and dispersion for the directional tuning of neuronal spike counts. *Under review*.

See Also

`kern.dpreg.circ`

Examples

```
data(spikes)
direction<-circular(spikes$direction,units="degrees")
counts<-spikes$counts
bw.joint.dpcirc(direction, counts, lower=c(0.5,0.5), upper=c(50,7))
```

---

bw.modalreg.circ.lin *Smoothing parameter selection for circular multimdodal regression*

Description

Function `bw.modalreg.circ.lin` provides the modal cross-validation smoothing parameters for the multimodal regression estimator when the covariate is circular and the response variable is linear.

Function `bw.modalreg.circ.circ` provides the modal cross-validation smoothing parameters for the multimodal regression estimator when the covariate and the response variable are circular.

Function `bw.modalreg.lin.circ` provides the modal cross-validation smoothing parameters for the multimodal regression estimator when the covariate is linear and the response variable is circular.

Usage

```
bw.modalreg.circ.lin(x, y, lower = NULL, upper = NULL, maxit = 500, tol = 0.00001)
bw.modalreg.circ.circ(x, y, lower = NULL, upper = NULL, maxit = 500, tol = 0.00001)
bw.modalreg.lin.circ(x, y, lower = NULL, upper = NULL, maxit = 500, tol = 0.00001)
```

Arguments

`x`	Vector of data for the independent variable. The object is coerced to class circular when using functions bw.modalreg.circ.lin and bw.modalreg.circ.circ.
`y`	Vector of data for the dependent variable. This must be same length as `x`. The object is coerced to class circular when using functions bw.modalreg.circ.circ and bw.modalreg.lin.circ.
`lower, upper`	Vectors of length two with the `lower` and `upper` boundaries of the intervals to be used in the search for the values of the smoothing parameters.
`maxit`	Maximum number of iterations in the estimation through the conditional (circular) mean shift.
`tol`	Tolerance parameter for convergence in the estimation through the conditional (circular) mean shift.

Details

See Alonso-Pena and Crujeiras (2022) for details.

The NAs will be automatically removed.

Value

A vector of length two with the first component being the value of the smoothing parameter associated to the predictor variable and with the second component being the value of the smoothing parameter associated to the response variable.

Author(s)

Maria Alonso-Pena and Rosa M. Crujeiras.

References

Alonso-Pena, M. and Crujeiras, R. M. (2022). Analizing animal escape data with circular nonparametric multimodal regression. Annals of Applied Statistics. (To appear).

See Also

modalreg.circ.lin, modalreg.circ.circ, modalreg.lin.circ

Examples

```
# Circ-lin
set.seed(8833)
n1<-100
n2<-100
gamma<-8
sigma<-1.5
theta1<-rcircularuniform(n1)
theta2<-rcircularuniform(n2)
theta<-c(theta1,theta2)
y1<-2*sin(2*theta1)+rnorm(n1,sd=sigma)
y2<-gamma+2*sin(2*theta2)+rnorm(n2,sd=sigma)
y<-as.numeric(c(y1,y2))
bw<-bw.modalreg.circ.lin(theta, y)

# Lin-circ
n1<-100
n2<-100
con<-8
set.seed(8833)
x1<-runif(n1)
x2<-runif(n2)
phi1<-(6*atan(2.5*x1-3)+rvonmises(n1,m=0,k=con))
phi2<-(pi+6*atan(2.5*x2-3)+rvonmises(n2,m=0,k=con))
x<-c(x1,x2)
phi<-c(phi1,phi2)
bw<-bw.modalreg.lin.circ(x, phi)

# Circ-circ
n1<-100
n2<-100
con<-10
```

```
set.seed(8833)
theta1<-rcircularuniform(n1)
theta2<-rcircularuniform(n2)
phi1<-(2*cos(theta1)+rvonmises(n1,m=0,k=con))
phi2<-(3*pi/4+2*cos(theta2)+rvonmises(n2,m=0,k=con))
theta=c(theta1,theta2)
phi=c(phi1,phi2)
bw<-bw.modalreg.lin.circ(theta, phi)
```

---

bw.pi *Plug–in rule*

Description

This function implements the von Mises scale plug-in rule for the smoothing parameter for density estimation when the number of components in the mixture is selected by Akaike Information Criterion (AIC) which selects the best model between a mixture of 2-5 von Mises distributions.

Usage

```
bw.pi(x, M=NULL, lower=0, upper=100, np=500, tol=0.1, outM=FALSE)
```

Arguments

x	Data from which the smoothing parameter is to be computed. The object is coerced to class circular.
M	Integer indicating the number of components in the mixture. If M=1, the rule of thumb is carried out with κ estimated by maximum likelihood. If M=NULL, AIC will be used.
lower, upper	lower and upper boundary of the interval to be used in the search for the value of the smoothing parameter. Default lower=0 and upper=100.
np	Number of points where to evaluate the estimator for numerical integration. Default np=500.
tol	Convergence tolerance for optimize. Default tol=0.1.
outM	Logical; if TRUE the function also returns the number of components in the mixture. Default, outM=FALSE.

Details

The value of the smoothing parameter is chosen by minimizing the asymptotic mean integrated squared error (AMISE) derived by Di Marzio et al. (2009) assuming that the data follow a mixture of von Mises distributions. The number of components in the mixture can be fixed by the user, by specifying the argument M or selected by using AIC (default option) as described in Oliveira et al. (2012). The NAs will be automatically removed.

Value

Vector with the value of the smoothing parameter and the number of components in the mixture (if specified).

Author(s)

Maria Oliveira, Rosa M. Crujeiras and Alberto Rodriguez–Casal

References

Oliveira, M., Crujeiras, R.M. and Rodriguez–Casal, A. (2012) A plug–in rule for bandwidth selection in circular density. *Computational Statistics and Data Analysis*, **56**, 3898–3908.

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2014) NPCirc: an R package for nonparametric circular methods. *Journal of Statistical Software*, **61**(9), 1–26. https://www.jstatsoft.org/v61/i09/

See Also

`kern.den.circ`, `bw.rt`, `bw.CV`, `bw.boot`

Examples

```
set.seed(2012)
n <- 100
x <- rcircmix(n,model=18)
bw.pi(x, M=3)
bw.pi(x, outM=TRUE)  # Using AIC
```

---

bw.reg.circ.lin — *Cross-validation rule for circular regression estimation*

Description

Function `bw.reg.circ.lin` provides the least squares cross-validation smoothing parameter for the Nadaraya-Watson and Local-Linear estimators when the covariate is circular and the response variable is linear.

Function `bw.reg.circ.circ` provides the least squares cross-validation smoothing parameter for the Nadaraya-Watson and Local-Linear estimators when the covariate and the response variable are circular.

Function `bw.reg.lin.circ` provides the least squares cross-validation smoothing parameter for the Nadaraya-Watson and Local-Linear estimators when the covariate is linear and the response variable is circular.

Usage

```
bw.reg.circ.lin(x, y, method="LL", lower=0, upper=50, tol=1e-2)
bw.reg.circ.circ(x, y, method="LL", option=1, lower=0, upper=50, tol=1e-2)
bw.reg.lin.circ(x, y, method="LL", option=1, lower=0, upper=50, tol=1e-2)
```

Arguments

`x`	Vector of data for the independent variable. The object is coerced to class `circular` when using functions `bw.reg.circ.lin` and `bw.reg.circ.circ`.
`y`	Vector of data for the dependent variable. This must be same length as `x`. The object is coerced to class `circular` when using functions `bw.reg.circ.circ` and `bw.reg.lin.circ`.
`method`	Character string giving the estimator to be used. This must be one of `"LL"` or `"NW"`. Default `method="LL"`.
`option`	Cross–validation rule. Default `option=1`. See details.
`lower, upper`	`lower` and `upper` boundary of the interval to be used in the search for the value of the smoothing parameter. Default `lower=0` and `upper=50`.
`tol`	Convergence tolerance for `optimize`. Default `tol=1e-2`.

Details

For nonparmetric regression with circular response, given (X_i, Y_i), $i = 1, \ldots, n$: If `option=1`, the cross–validation smoothing parameter is computed as the value that minimizes $\sum_{i=1}^{n}(-\cos(Y_i - \hat{f}^{-i}(X_i))$, where $\hat{f}^{-i}$ denotes the estimator computed with all the observations except (X_i, Y_i).

If `option=2`, the cross–validation smoothing parameter is computed as the value that minimizes $n^{-1}\sum_{i=1}^{n}(d(Y_i, \hat{f}^{-i}(X_i))^2$ where $d(Y_i, \hat{f}^{-i}(X_i) = \min(|Y_i - \hat{f}^{-i}(X_i)|, 2\pi - |Y_i - \hat{f}^{-i}(X_i)|)$.

The NAs will be automatically removed.

Value

Value of the smoothing parameter.

Author(s)

Maria Oliveira, Rosa M. Crujeiras and Alberto Rodriguez–Casal

References

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2013) Nonparametric circular methods for exploring environmental data. *Environmental and Ecological Statistics*, **20**, 1–17.

Di Marzio, M., Panzera A. and Taylor, C. C. (2012) Non–parametric regression for circular responses. *Scandinavian Journal of Statistics*, **40**, 228–255.

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2014) NPCirc: an R package for nonparametric circular methods. *Journal of Statistical Software*, **61**(9), 1–26. https://www.jstatsoft.org/v61/i09/

See Also

`kern.reg.circ.lin`, `kern.reg.circ.circ`, `kern.reg.lin.circ`

Examples

```
set.seed(2012)
n <- 100
x <- seq(0,2*pi,length=n)
y <- sin(x)+0.2*rnorm(n)
bw.reg.circ.lin(circular(x), y, method="LL", lower=1, upper=20)
bw.reg.circ.lin(circular(x), y, method="NW", lower=1, upper=20)
```

---

bw.rt *Rule of thumb*

Description

This function implements the selector proposed by Taylor (2008) for density estimation, based on an estimation of the concentration parameter of a von Mises distribution. The concentration parameter can be estimated by maximum likelihood or by a robustified procedure as described in Oliveira et al. (2013).

Usage

```
bw.rt(x, robust=FALSE, alpha=0.5)
```

Arguments

`x`	Data from which the smoothing parameter is to be computed. The object is coerced to class `circular`.
`robust`	Logical, if `robust=FALSE` the parameter κ is estimated by maximum likelihood, if `TRUE` it is estimated as described in Oliveira et al. (2012b). Default `robust=FALSE`.
`alpha`	Arc probability when `robust=TRUE`. Default is `alpha=0.5`. See Details.

Details

When `robust=TRUE`, the parameter κ is estimated as follows:

1. Select $\alpha \in (0, 1)$ and find the shortest arc containing $\alpha \cdot 100\%$ of the sample data.

2. Obtain the estimated $\hat{\kappa}$ in such way that the probability of a von Mises centered in the midpoint of the arc is `alpha`.

The NAs will be automatically removed.

See also Oliveira et al. (2012).

Value

Value of the smoothing parameter.

Author(s)

Maria Oliveira, Rosa M. Crujeiras and Alberto Rodriguez–Casal

References

Oliveira, M., Crujeiras, R.M. and Rodriguez–Casal, A. (2012) A plug–in rule for bandwidth selection in circular density. *Computational Statistics and Data Analysis*, **56**, 3898–3908.

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2013) Nonparametric circular methods for exploring environmental data. *Environmental and Ecological Statistics*, **20**, 1–17.

Taylor, C.C. (2008) Automatic bandwidth selection for circular density estimation. *Computational Statistics and Data Analysis*, **52**, 3493–3500.

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2014) NPCirc: an R package for nonparametric circular methods. *Journal of Statistical Software*, **61**(9), 1–26. https://www.jstatsoft.org/v61/i09/

See Also

kern.den.circ, bw.CV, bw.pi, bw.boot

Examples

```
set.seed(2012)
n <- 100
x <- rcircmix(n,model=7)
bw.rt(x)
bw.rt(x, robust=TRUE)
```

---

circ.local.lik — *Local likelihood estimation for regression with circular covariate*

Description

Function circ.local.lik computes a nonparametric estimation of a curve of interest, regarded as a transformation of the mean regression function, when the predictor is circular and the conditional density is either gaussian, Bernoulli, Poisson or gamma. It also computes the derivatives of the function of interest. It uses the method described in Alonso-Pena et al. (2022).

Usage

```
circ.local.lik(x, y, t = NULL, bw = NULL, family, p = 1,
  startv = NULL, tol = 0.00001, maxit = 300, from = circular(0),
  to = circular(2 * pi),len = 250)
```

Arguments

x	Vector of data for the independent variable. The object is coerced to class circular.
y	Vector of data for the dependent variable. This must be same length as x.
t	Points where the regression function is estimated. If NULL, equally spaced points are used according to the parameters from, to and len.

bw	Value of the smoothing (concentration) parameter used. The value of the smoothing parameter can be chosen by using `bw.circ.local.lik`.
family	Character string indicating the conditional density to be used. It must be one of `"gaussian"`, `"bernoulli"`, `"poisson"` or `"gamma"`. When `family = "gaussian"`, the conditional mean is estimated; when `family = "bernoulli"`, the logit function is estimated and when `family = "poisson"` or `family = "gamma"`, the function of interest is the logarithm of the conditional mean.
p	Degree of the local sine-polynomial to be used in the estimation process. It must be 1 or 3.
startv	Vector containing the initial values for the estimation algorithm if family is set as `"bernoulli"`, `"poisson"` or `"gamma"`. The vector must be of length 2 if `p = 1` and length 4 if `p = 3`. If `NULL`, the initial parameters are the ones corresponding to the global mean of the responses.
tol	Tolerance parameter for convergence in the numerical estimation. Only needed if family is one of `"bernoulli"`, `"poisson"` or `"gamma"`. Default is `tol = 0.00001`.
maxit	Maximum number of iterations in the numerical estimation. Only needed if family is one of `"bernoulli"`, `"poisson"` or `"gamma"`. Default is `maxit = 300`.
from, to	Left and right-most points of the grid at which the density is to be estimated. The objects are coerced to class `circular`.
len	Number of equally spaced points at which the density is to be estimated.

Details

See Alonso-Pena et al. (2022).

The NAs will be automatically removed.

Value

A list containing the following components:

datax, datay	Original dataset.
x	The n coordinates of the points where the regression function and its derivatives are estimated.
y	A list containing the estimated values of the function of interest and its derivatives up to order `p`.
bw	The smoothing parameter used.
n	The sample size after elimination of missing values.
call	The call which produced the result.
data.name	The deparsed name of the x argument.
has.na	Logical, for compatibility (always FALSE).

Author(s)

Maria Alonso-Pena, Irene Gijbels and Rosa M. Crujeiras.

References

Alonso-Pena, M., Gijbels, I. and Crujeiras, R.M. (2022). A general framework for circular local likelihood regression. *Under review*.

See Also

bw.circ.local.lik

Examples

```
data(spikes)
direction<-circular(spikes$direction,units="degrees")
counts<-spikes$counts
circ.local.lik(direction, counts, bw=8, p=1, family="poisson")
```

---

circsizer.density *CircSiZer map for density*

Description

This function plots the CircSiZer map for circular density estimation based on circular kernel methods, as described in Oliveira et al. (2013). The CircSiZer is an extension of SiZer proposed by Chaudhuri and Marron (1999) to circular data.

Usage

```
circsizer.density(x, bws, ngrid=250, alpha=0.05, B=500, log.scale=TRUE,
display=TRUE)
## S3 method for class 'circsizer'
print(x, digits=NULL, ...)
```

Arguments

`x`	Data from which the estimate is to be computed. The object is coerced to class `circular`.
`bws`	Vector of smoothing parameters. Values of `bws` must be positive. `bws` will be coerced to be equally spaced. Length of vector `bws` must be at least 2.
`ngrid`	Integer indicating the number of equally spaced angles between 0 and 2π where the estimator is evaluated. Default `ngrid=250`.
`alpha`	Significance level for the CircSiZer map. Default `alpha=0.05`.
`B`	Integer indicating the number of bootstrap samples to estimate the standard deviation of the derivative estimator. Default `B=500`.
`log.scale`	Logical, if `TRUE`, the CircSiZer map is plotted in the scale $-\log_{10}$(`bws`). Default is `TRUE`. See Details.
`display`	Logical, if `TRUE`, the CircSiZer map is plotted. Default is `TRUE`.
`digits`	Integer indicating the precision to be used.
`...`	further arguments

Details

With CircSiZer, significance features (peaks and valleys) in the data are seeked via the construction of confidence intervals for the scale-space version of the smoothed derivative curve, as it is described in Oliveira et al. (2013). Thus, for a given point and a given value of the smoothing parameter, the curve is significantly increasing (decreasing) if the confidence interval is above (below) 0 and if the confidence interval contains 0, the curve for that value of the smoothing parameter and at that point does not have a statistically significant slope. If `display=TRUE`, this information is displayed in a circular color map, the CircSiZer map, in such a way that, at a given point, the performance of the estimated curve is represented by a color ring with radius proportional to the value of the smoothing parameter.

Differents colors allow to indentify peaks and valleys. Blue color indicates locations where the curve is significantly increasing; red color shows where it is significantly decreasing and purple indicates where it is not significantly different from zero. Gray color corresponds to those regions where there is not enough data to make statements about significance. Thus, at a given bandwidth, a significant peak can be identified when a region of significant positive gradient is followed by a region of significant negative gradient (i.e. blue-red pattern), and a significant trough by the reverse (red-blue pattern), taking clockwise as the positive sense of rotation.

If `log.scale=TRUE` then, the values of the considered smoothing parameters `bws` are transformed to $-\log_{10}$ scale, i.e, a sequence of equally spaced smoothing parameters according to the parameters `-log10(max(bws))`, `-log10(min(bws))` and `length(bws)` is used. Hence, small values of this parameter corresponds with larger rings and large values corresponds with smaller rings. Whereas if `log.scale=FALSE`, small values of this parameter corresponds with smaller rings and large values corresponds with larger rings.

The NAs will be automatically removed.

Value

An object with class `circsizer` whose underlying structure is a list containing the following components:

`data`	Original dataset.
`ngrid`	Number of equally spaced angles where the derivative of the circular kernel density estimator.
`bw`	Vector of smoothing parameters (given in $-\log_{10}$ scale if `log.scale=TRUE`).
`log.scale`	Logical; if `TRUE`, the $-\log_{10}$ scale is used for constructing the CircSiZer map.
`CI`	List containing: a matrix with lower limits fot the confidence intervals; a matrix with the lower limits of the confidence intervals; a matrix with the Effective Sample Size. Each row corresponds to each value of the smoothing parameter and each column corresponds to an angle.
`col`	Matrix containing the colors for plotting the CircSiZer map.

If `display==TRUE`, the function also returns the CircSiZer map for density.

Author(s)

Maria Oliveira, Rosa M. Crujeiras and Alberto Rodriguez–Casal

References

Chaudhuri, P. and Marron, J.S. (1999). SiZer for exploration of structures in curves, *Journal of the American Statistical Association*, **94**, 807–823.

Oliveira, M., Crujeiras, R.M. and Rodriguez–Casal (2014) CircSiZer: an exploratory tool for circular data. *Environmental and Ecological Statistics*, **21**, 143–159.

Oliveira, M., Crujeiras R.M. and Rodr?guez–Casal, A. (2014) NPCirc: an R package for nonparametric circular methods. *Journal of Statistical Software*, **61**(9), 1–26. https://www.jstatsoft.org/v61/i09/

See Also

circsizer.map

Examples

```
## Not run:
set.seed(2012)
x <- rcircmix(100,model=7)
sizer <- circsizer.density(x, bws=seq(0,50,length=12))
sizer
names(sizer)
circsizer.map(sizer,type=1,zero=pi/2,clockwise=TRUE,raw.data=TRUE)

## End(Not run)
```

---

circsizer.map *CircSiZer map*

Description

This function plots the CircSiZer map for `circsizer` objects.

Usage

```
circsizer.map(circsizer.object, type, zero, clockwise, title=NULL, labels=NULL,
label.pos=NULL, rad.pos=NULL, raw.data=FALSE)
```

Arguments

`circsizer.object`	An object of class `circsizer`, i.e., output from functions `circsizer.density` or `circsizer.regression`.
`type`	Number indicating the labels to display in the plot: 1 (directions), 2 (hours), 3 (angles in radians), 4 (angles in degrees) or 5 (months).
`zero`	Where to place the starting (zero) point.
`clockwise`	Whether to interpret positive positions as clockwise from the starting point.

`title`	Title for the plot.
`labels`	Character or expression vector of labels to be placed at the `label.pos`. `label.pos` must also be supplied.
`label.pos`	Vector indicating the position (between 0 and 2π) at which the labels are to be drawn.
`rad.pos`	Vector (between 0 and 2π) with the drawing position for the radius.
`raw.data`	Logical, if `TRUE`, points indicated by `x` are stacked on the perimeter of the circle. Default is `FALSE`.

Value

CircSiZer map.

Author(s)

Maria Oliveira, Rosa M. Crujeiras and Alberto Rodriguez–Casal

References

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2014) NPCirc: an R package for nonparametric circular methods. *Journal of Statistical Software*, **61**(9), 1–26. https://www.jstatsoft.org/v61/i09/

See Also

`circsizer.density`, `circsizer.regression`

`circsizer.regression` *CircSiZer map for regression*

Description

This function plots the CircSiZer map for circular regression estimation based on circular kernel methods, as described in Oliveira et al. (2013). The CircSiZer is an extension of SiZer proposed by Chaudhuri and Marron (1999) to circular data.

Usage

```
circsizer.regression(x, y, bws=NULL, adjust=2, ngrid=150, alpha=0.05, B=500,
B2=250, log.scale=TRUE, display=TRUE)
```

Arguments

`x`	Vector of data for the independent variable. The object is coerced to class `circular`.
`y`	Vector of data for the dependent variable. This must be same length as `x`.
`bws`	Vector of smoothing parameters. Values of `bws` must be positive. `bws` will be coerced to be equally spaced. Length of vector `bws` must be at least 2.
`adjust`	If `bws=NULL`, the smoothing parameters used are `adjust/bw` and `adjust*bw`, where `bw` is the smoothing parameter obtained by using the cross–validation rule.
`ngrid`	Integer indicating the number of equally spaced angles between 0 and 2π where the estimator is evaluated. Default `ngrid=150`.
`alpha`	Significance level for the CircSiZer map. Default `alpha=0.05`.
`B`	Integer indicating the number of bootstrap samples to estimate the standard deviation of the derivative estimator. Default `B=500`.
`B2`	Integer indicating the number of bootstrap samples to compute the denominator in Step 2 of algorithm described in Oliveira et al. (2013). Default `B=250`.
`log.scale`	Logical, if `TRUE`, the CircSiZer map is plotted in the scale $-\log_{10}($`bws`$)$. Default is `TRUE`.
`display`	Logical, if `TRUE`, the CircSiZer map is plotted. Default is `TRUE`.

Details

See Details Section of `circsizer.density`. The NAs will be automatically removed.

Value

An object with class `circsizer` whose underlying structure is a list containing the following components.

`data`	Original dataset.
`ngrid`	Number of equally spaced angles where the derivative of the regression estimator is evaluated.
`bw`	Vector of smoothing parameters (given in $-\log_{10}$ scale if `log.scale=TRUE`).
`log.scale`	Logical; if `TRUE`, the $-\log_{10}$ scale is used for constructing the CircSiZer map.
`CI`	List containing: a matrix with lower limits fot the confidence intervals; a matrix with the lower limits of the confidence intervals; a matrix with the Effective Sample Size. Each row corresponds to each value of the smoothing parameter and each column corresponds to an angle.
`col`	Matrix containing the colors for plotting the CircSiZer map.

If `display==TRUE`, the function also returns the CircSiZer map for regression.

Author(s)

Maria Oliveira, Rosa M. Crujeiras and Alberto Rodriguez–Casal

References

Chaudhuri, P. and Marron, J.S. (1999). SiZer for exploration of structures in curves, *Journal of the American Statistical Association*, **94**, 807–823.

Oliveira, M., Crujeiras, R.M. and Rodriguez–Casal (2014) CircSiZer: an exploratory tool for circular data. *Environmental and Ecological Statistics*, **21**, 143–159.

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2014) NPCirc: an R package for nonparametric circular methods. *Journal of Statistical Software*, **61**(9), 1–26. https://www.jstatsoft.org/v61/i09/

See Also

circsizer.map

Examples

```
## Not run:
set.seed(2012)
n <- 100
x <- seq(0,2*pi,length=n)
y <- sin(x)+sqrt(0.5)*rnorm(n)
circsizer.regression(circular(x), y, bws=seq(10,60,by=5))

## End(Not run)
```

---

cross.beds1 *Cross-beds azimuths (I)*

Description

This dataset corresponds to azimuths of cross-beds in the Kamthi river (India). Originally analized by SenGupta and Rao (1966) and included in Table 1.5 in Mardia (1972), the dataset collects 580 azimuths of layers lying oblique to principal accumulation surface along the river, being these layers known as cross-beds.

Usage

```
data(cross.beds1)
```

Format

A single-column data frame with 580 observations in radians.

Details

Data were originally recorded in degrees.

Source

Mardia, K.V. (1972) *Statistics of Directional Data*. Academic Press, New York.

SenGupta, S. and Rao, J.S. (1966) Statistical analysis of cross–bedding azimuths from the Kamthi formation around Bheemaram, Pranhita: Godavari Valley. *Sankhya: The Indian Journal of Statistics, Series B*, **28**, 165–174.

Examples

```
data(cross.beds1)
```

`cross.beds2` *Cross-beds (II)*

Description

A dataset of cross-beds measurements from Himalayan molasse in Pakistan presented in Fisher (1993). This dataset collects 104 measurements of Chaudan Zam large bedforms.

Usage

```
data(cross.beds2)
```

Format

A single-column data frame with 104 observations in radians.

Details

Data were originally recorded in degrees.

Source

Fisher, N.I. (1993) *Statistical Analysis of Circular Data*. Cambridge University Press, Cambridge, U.K.

Examples

```
data(cross.beds2)
```

`cycle.changes` *Cycle changes*

Description

The data consists on the changes in cycles of temperatures at ground level in periglacial Monte Alvear (Argentina). The dataset includes 350 observations which correspond to the hours where the temperature changes from positive to negative and viceversa from February 2008 to December 2009.

Usage

```
data(cycle.changes)
```

Format

A data frame with 350 observations on two variables: `change`, which indicates if the temperature changed from positive to negative (-1) or viceversa (1) and `hour`, which indicates the hour (in radians) when the cycle change occured.

Details

Analysis of cycle changes in temperatures for another locations can be seen in Oliveira et al. (2013).

Source

The authors want to acknowledge Prof. Augusto Perez–Alberti for providing the data, collected within the Project POL2006-09071 from the Spanish Ministry of Education and Science.

References

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2013) Nonparametric circular methods for exploring environmental data. *Environmental and Ecological Statistics*, **20**, 1–17.

Examples

```
data(cycle.changes)
thaw <- (cycle.changes[,1]==1)
frosting <- (cycle.changes[,1]==-1)
plot(circular(cycle.changes[frosting,2],template="clock24"), shrink=1.08, col=4,
stack=TRUE, main="Frosting")
plot(circular(cycle.changes[thaw,2],template="clock24"), shrink=1.08, col=2,
stack=TRUE, main="Thaw")
```

`dcircmix` *Mixtures of circular distributions*

Description

Density and random generation functions for a circular distribution or a mixture of circular distributions allowing the following components: circular uniform, von Mises, cardioid, wrapped Cauchy, wrapped normal, wrapped skew-normal.

Usage

```
dcircmix(x, model=NULL, dist=NULL, param=NULL)
rcircmix(n, model=NULL, dist=NULL, param=NULL)
```

Arguments

`x`	Vector of angles where the density is evaluated. The object is coerced to class `circular`.
`n`	Number of observations to generate.
`model`	Number between 1 and 20, corresponding with a model defined in Oliveira et al. (2012). See Details.
`dist`	Vector of strings with the distributions that participate in the mixture: `"unif"`, `"vm"`, `"car"`, `"wc"`, `"wn"`, `"wsn"`.
`param`	List with three or four objects. The first object will be a vector containing the proportion of each distribution in the mixture, the second object will be a vector containg the location parameters and the third object will be a vector containing the concentration parameters. If the wrapped skew-normal distribution participates in the mixture, a fourth object will be introduced in the list, a vector containing the skewness parameter. In this case, the values of the skewness parameter for the rest of distributions in the mixture will be zero. The length of each object in the list must be equal to the length of argument `dist`. See Details and Examples.

Details

Models from Oliveira et al. (2012) are described below:

M1: Circular uniform.

M2: von Mises: $vM(\pi, 1)$.

M3: Wrapped normal: $WN(\pi, 0.9)$.

M4: cardioid: $C(\pi, 0.5)$.

M5: Wrapped Cauchy: $WC(\pi, 0.8)$.

M6: Wrapped skew–normal: $WSN(\pi, 1, 20)$.

M7: Mixture of two von Mises $1/2vM(0, 4) + 1/2vM(\pi, 4)$.

M8: Mixture of two von Mises $1/2vM(2,5) + 1/2vM(4,5)$.

M9: Mixture of two von Mises $1/4vM(0,2) + 3/4vM(\pi/\sqrt{3},2)$.

M10: Mixture of von Mises and wrapped Cauchy $4/5vM(\pi,5) + 1/5WC(4\pi/3,0.9)$.

M11: Mixture of three von Mises $1/3vM(\pi/3,6) + 1/3vM(\pi,6) + 1/3vM(5\pi/3,6)$.

M12: Mixture of three von Mises $2/5vM(\pi/2,4) + 1/5vM(\pi,5) + 2/5vM(3\pi/2,4)$.

M13: Mixture of three von Mises $2/5vM(0.5,6) + 2/5vM(3,6) + 1/5vM(5,24)$.

M14: Mixture of four von Mises $1/4vM(0,12)+1/4vM(\pi/2,12)+1/4vM(\pi,12)+1/4vM(3\pi/2,12)$.

M15: Mixture of wrapped Cauchy, wrapped normal, von Mises and wrapped skew-normal $3/10WC(\pi-1,0.6) + 1/4WN(\pi+0.5,0.9) + 1/4vM(\pi+2,3) + 1/5WSN(6,1,3)$.

M16: Mixture of five von Mises $1/5vM(\pi/5,18)+1/5vM(3\pi/5,18)+1/5vM(\pi,18)+1/5vM(7\pi/5,18)+1/5vM(9\pi/5,18)$.

M17: Mixture of cardioid and wrapped Cauchy $2/3C(\pi,0.5) + 1/3WC(\pi,0.9)$.

M18: Mixture of four von Mises $1/2vM(\pi,1)+1/6vM(\pi-0.8,30)+1/6vM(\pi,30)+1/vM(\pi+0.8,30)$.

M19: Mixture of five von Mises $4/9vM(2,3)+5/36vM(4,3)+5/36vM(3.5,50)+5/36vM(4,50)+5/36vM(4.5,50)$.

M20: Mixture of two wrapped skew-normal and two wrapped Cauchy $1/3WSN(0,0.7,20) + 1/3WSN(\pi,0.7,20) + 1/6WC(3\pi/4,0.9) + 1/6WC(7\pi/4,0.9)$.

When the wrapped skew-normal distribution participates in the mixture, the argument `param` for function `dcircmix` can be a list with fifth objects. The fifth object would be the number of terms to be used in approximating the density function of the wrapped skew normal distribution. By default the number of terms used is 20.

Value

`dcircmix` gives the density and `rcircmix` generates random deviates.

Author(s)

Maria Oliveira, Rosa M. Crujeiras and Alberto Rodriguez–Casal

References

Oliveira, M., Crujeiras, R.M. and Rodriguez–Casal, A. (2012) A plug–in rule for bandwidth selection in circular density. *Computational Statistics and Data Analysis*, **56**, 3898–3908.

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2014) NPCirc: an R package for nonparametric circular methods. *Journal of Statistical Software*, **61**(9), 1–26. https://www.jstatsoft.org/v61/i09/

Examples

```
set.seed(2012)
# Circular representation of models M1-M20, each one in a separate window
for (i in 1:20){
dev.new()
```

```
f <- function(x) dcircmix(x, model=i)
curve.circular(f, n=500, join=TRUE, shrink=1.9, main=i)
}

# Random generation from model M1 (uniform model)
data1 <- rcircmix(50, model=1)
plot(data1)

# Density function and random generation from a mixture of a von Mises and
# a wrapped skew-normal
f <- function(x) dcircmix(x, model=NULL, dist=c("vm","wsn"),
param=list(p=c(0.5,0.5), mu=c(0,pi), con=c(1,1), sk=c(0,10)))
curve.circular(f, n=500, shrink=1.2)
data <- rcircmix(100, model=NULL, dist=c("vm","wsn"),
param=list(p=c(0.5,0.5), mu=c(0,pi), con=c(1,1), sk=c(0,10)))
points(data)

# Density function and random generation from a mixture of two von Mises and
# two wrapped Cauchy
f <- function(x) dcircmix(x, model=NULL, dist=c("vm","vm","wc","wc"),
param=list(p=c(0.3,0.3,0.2,0.2), mu=c(0,pi,pi/2,3*pi/2), con=c(5,5,0.9,0.9)))
curve.circular(f, n=1000, xlim=c(-1.65,1.65))
data <- rcircmix(100, model=NULL, dist=c("vm","vm","wc","wc"),
param=list(p=c(0.3,0.3,0.2,0.2), mu=c(0,pi,pi/2,3*pi/2), con=c(5,5,0.9,0.9)))
points(data)
```

`dpreg.circ` *Joint parametric estimation of mean and dispersion functions in circular double Poisson models*

Description

Function `dpreg.circ` implements the parametric joint estimator of the mean and dispersion functions when the covariate is circular and the conditional distribution is a double Poisson, a particular case of the double exponential family. It is assumed that the logarithm of the mean and the logarithm of the dispersion are sums of sine and cosine terms.

Usage

```
dpreg.circ(x, y, k = 2, ktilde = 1, startvmu = NULL, startvgam = NULL,
          tol= 0.000001, maxit = 300)
```

Arguments

x	Vector of data for the independent variable. The object is coerced to class `circular`.
y	Vector of data for the dependent variable. This must be same length as x and should contain counts.

k	Number of components for modeling the logarithm of the mean, including the intercept. Equivalent to the number of parameters to be estimated for the mean function.
ktilde	Number of components for modeling the logarithm of the dispersion, including the intercept. Equivalent to the number of parameters to be estimated for the dispersion function.
startvmu	Vector of length k containing the initial values for the parameters corresponding to the estimation of the mean.
startvgam	Vector of length ktilde containing the initial values for the parameters corresponding to the estimation of the dispersion.
tol	Tolerance parameter for convergence in the numerical estimation.
maxit	Maximum number of iterations in the numerical estimation.

Details

See Alonso-Pena et al. (2022) for details.

Value

A list containing the following components:

datax, datay	Original dataset.
coefficients_mu	A vector of length k containing the estimators for the parameters corresponding to the mean.
coefficients_mu	A vector of length ktilde containing the estimators for the parameters corresponding to the dispersion.
numit	Number of iterations needed for convergence.
n	The sample size after elimination of missing values.
call	The call which produced the result.
data.name	The deparsed name of the x argument.
has.na	Logical, for compatibility (always FALSE).

Author(s)

Maria Alonso-Pena, Irene Gijbels and Rosa M. Crujeiras

References

Alonso-Pena, M., Gijbels, I. and Crujeiras, R.M. (2022). Flexible joint modeling of mean and dispersion for the directional tuning of neuronal spike counts. *Under review*.

Examples

```
data(spikes)
direction<-circular(spikes$direction,units="degrees")
counts<-spikes$counts
output<-dpreg.circ(direction, counts,  k = 5, ktilde = 3)
```

`dragonfly` *Orientations of dragonflies*

Description

The data, presented in Batschelet (1981), consists on the orientation of 214 dragonflies with respect to the sun's azimuth.

Usage

```
data(dragonfly)
```

Format

A single-column data frame with 214 observations in radians.

Details

Data were originally recorded in degrees.

Source

Batschelet, E. (1981) *Circular Statistics in Biology*. Academic Press, New York.

Examples

```
data(dragonfly)
x <- circular(dragonfly$orientation)
dens <- kern.den.circ(x)
plot(dens, shrink=1.3)
```

dwsn *Wrapped skew-Normal density function*

Description

Density function and random generation for the wrapped skew-Normal distribution introduced by Pewsey (2000).

Usage

```
dwsn(x,xi,eta,lambda,K=NULL,min.k=20)
rwsn(n,xi,eta,lambda)
```

Arguments

`x`	Vector of angles where the density is evaluated . The object is coerced to class `circular`.
`n`	Number of observations.
`xi`	Location parameter. The object is coerced to class `circular`.
`eta`	Scale parameter.
`lambda`	Skewness parameter.
`K`	Number of terms to be used in approximating the density. Default `K=NULL`.
`min.k`	Minimum number of terms used in approximating the density.

Details

The NAs will be automatically removed.

Value

`dwsn` gives the density and `rwsn` generates random deviates.

Author(s)

Maria Oliveira, Rosa M. Crujeiras and Alberto Rodriguez–Casal

References

Pewsey, A. (2000) The wrapped skew–Normal distribution on the circle. *Communications in Statistics - Theory and Methods*, **29**, 2459–2472.

Examples

```
set.seed(2012)
# Density function of a wrapped skew-normal distribution WSN(pi,1,20)
wsn <- function(x) dwsn(x, xi=circular(pi), eta=1, lambda=20)
curve.circular(wsn,n=500,xlim=c(-1.65,1.65),main=expression(WSN(pi,1,20)))
# Random generation
data<-rwsn(50,xi=circular(pi),eta=1,lambda=20)
points(data)
```

`flywheels` *Flywheel measurements*

Description

The data consists on measurements on mechanical flywheels, which are tools to regulate an engine's rotation in the automotive industry. The dataset contains 60 observations, containing the flywheels' angle of imbalance and the magnitude required to correct them. Four different types of metals were employed in the production process.

Usage

```
data("flywheels")
```

Format

A data frame with 60 observations on the following 3 variables.

`angle` A numeric vector containing the angles of imbalance (in radians)

`weight` A numeric vector containing the magnitudes of correction (in inch-ounces).

`group` A vector with numbers from 1 to 4 indicating the type of metal.

Details

The data were analyzed in Anderson-Cook (1999) and Alonso-Pena et al. (2021).

Source

The dataset was obtained from Anderson-Cook (1999).

References

Anderson-Cook, C.M. (1999). A tutorial on one-way analysis of circular-linear data. *Journal of Quality Technology*, **31**(1), 109–119.

Alonso-Pena, M., Ameijeiras-Alonso, J. and Crujeiras, R.M. (2021) Nonparametric tests for circular regression. *Journal of Statistical Computation and Simulation*, **91**(3), 477–500.

Examples

```
data(flywheels)
```

`HumanMotorResonance` *Human motor resonance data*

Description

This dataset, given by Puglisi et al. (2017), contains measurements of motor resonance responses in humans, who were requested to observe a movement of a rythmic hand flexion-extension. The dataset is composed of 70 observations, including the angular position of the hand and the resonance response.

Usage

```
data("HumanMotorResonance")
```

Format

A data frame with 70 observations on the following 2 variables.

`Angular.position` A numeric vector containing the angular position of the hand (in radians).

`Reflex.Amplitude` A numeric vector containing the resonance response.

Details

See Puglisi et al. (2017) for details.

Source

The dataset was obtained by Puglisi et al. (2017).

References

Puglisi, G., Leonetti, A., Landau, A., Fornia, L., Cerri, G. and Borroni, P. (2017). The role of attention in human motor resonance. *PLOS ONE*, **12**(5), e0177457.

Examples

```
data(HumanMotorResonance)
```

kern.den.circ *Kernel density derivative estimate for circular data*

Description

This function computes the kernel density derivative estimate with the given kernel and bandwidth for circular data.

Usage

```
kern.den.circ(x,z=NULL,bw="AA",deriv.order=0,kernel="vonmises",na.rm = FALSE,
        from = circular(0), to = circular(2 * pi),n = 512,control.circular=list())
```

Arguments

- `x` — Data from which the estimate is to be computed. The object is coerced to class `circular`.
- `z` — Points where the density derivative is estimated. If `NULL` equally spaced points are used according to the parameters `from`, `to` and `n`.
- `bw` — Smoothing parameter to be used. `bw` can also be a character string giving a rule to choose the bandwidth. See `bw.AA`, `bw.pi`, `bw.rt`, `bw.CV`, and `bw.boot`. The default, `'AA'`, is the 2-stage solve-the-equation plug-in smoothing selector.
- `deriv.order` — Derivative order. Default `deriv.order=0` (density estimation).
- `kernel` — a character string giving the smoothing kernel to be used. This must be one of `"vonmises"` or `"wrappednormal"`.
- `na.rm` — logical; if `TRUE`, missing values are removed from `x`. If `FALSE` any missing values cause an error.
- `from, to` — the left and right-most points of the grid at which the density is to be estimated. The objects are coerced to class `circular`.
- `n` — the number of equally spaced points at which the density is to be estimated.
- `control.circular` — the attribute of the resulting objects (`x` component).

Value

An object with class `density.circular` whose underlying structure is a list containing the following components.

- `data` — original dataset.
- `x` — the `n` coordinates of the points where the density is estimated. It is a circular objects with coordinate system setting using `control.circular`.
- `y` — the estimated density values.
- `bw` — the smoothing parameter used.
- `N` — the sample size after elimination of missing values.

`call`	the call which produced the result.
`data.name`	the deparsed name of the `x` argument.
`has.na`	logical, for compatibility (always FALSE).

Author(s)

Jose Ameijeiras-Alonso.

References

Ameijeiras-Alonso, J. (2022) A reliable data-based smoothing parameter selection method for circular kernel estimation.

Di Marzio, M., Panzera, A., & Taylor, C. C. (2011). Kernel density estimation on the torus. *Journal of Statistical Planning and Inference*, **141**(6), 2156–2173.

Oliveira, M., Crujeiras R.M. and Rodr?guez-Casal, A. (2014) NPCirc: an R package for nonparametric circular methods. *Journal of Statistical Software*, **61**(9), 1–26. https://www.jstatsoft.org/v61/i09/

See Also

`bw.AA`, `plot.density.circular`, `lines.density.circular`, `bw.pi`, `bw.rt`, `bw.CV`, `bw.boot`

Examples

```
set.seed(2022)
n <- 50
x <- rcircmix(n, model=13)
# Using the smoothing parameter by default,
# i.e., 2-stage solve-the-equation plug-in rule
est1 <- kern.den.circ(x,deriv.order=1)
# Selecting the smoothing parameter: 2-stage direct plug-in rule
est2 <- kern.den.circ(x, bw="dpi", deriv.order=1)
# Circular plot
plot(est1, plot.type="circle", points.plot=TRUE, shrink=1.4,
      main="Circular plot",ylab="Density derivative circular")
lines(est2, plot.type="circle", shrink=1.4 ,col=2)
# Linear plot
plot(est1, plot.type="line", main="Linear plot",ylab="Density derivative circular")
lines(est2, plot.type="line", col=2)
```

`kern.dpreg.circ` *Joint kernel estimation of mean and dispersion functions in circular double Poisson models*

Description

Function `kern.dpreg.circ` implements the nonparametric joint estimator of the mean and dispersion functions when the covariate is circular and the conditional distribution is a double Poisson, a particular case of the double exponential family. It takes the von Mises distribution as the kernel employed for both the estimation of the mean and the dispersion. The employed estimator is a local-linear type.

Usage

```
kern.dpreg.circ(x, y, t = NULL, bw, startvmu = NULL, startvgam = NULL, tol= 0.000001,
  maxit = 300, from =     circular(0),to = circular(2 * pi), len = 250)
```

Arguments

`x`	Vector of data for the independent variable. The object is coerced to class `circular`.
`y`	Vector of data for the dependent variable. This must be same length as `x` and should contain counts.
`t`	Points where the regression function is estimated. If `NULL`, equally spaced points are used according to the parameters `from`, `to` and `len`.
`bw`	Numeric vector of length two containing the smoothing (concentration) parameters used. The first component is the parameter used for the estimation of the mean, while the second component is used for estimating the dispersion. The value of the smoothing parameters can be chosen by using `bw.joint.dpcirc`.
`startvmu`	Vector of length two containing the initial values for the parameters corresponding to the estimation of the mean.
`startvgam`	Vector of length two containing the initial values for the parameters corresponding to the estimation of the dispersion.
`tol`	Tolerance parameter for convergence in the numerical estimation.
`maxit`	Maximum number of iterations in the numerical estimation.
`from, to`	Left and right-most points of the grid at which the density is to be estimated. The objects are coerced to class `circular`.
`len`	Number of equally spaced points at which the function is to be estimated.

Details

See Alonso-Pena et al. (2022) for details.

The NAs will be automatically removed.

Value

A list containing the following components:

`datax, datay`	Original dataset.
`x`	The n coordinates of the points where the regression function and its derivatives are estimated.

estim	A list containing the estimated values of the logarithm of the mean function (first component) and the logarithm of the dispersion function (second component).
bw	The vector of smoothing parameters used.
n	The sample size after elimination of missing values.
call	The call which produced the result.
data.name	The deparsed name of the x argument.
has.na	Logical, for compatibility (always FALSE).

Author(s)

Maria Alonso-Pena, Irene Gijbels and Rosa M. Crujeiras

References

Alonso-Pena, M., Gijbels, I. and Crujeiras, R.M. (2022). Flexible joint modeling of mean and dispersion for the directional tuning of neuronal spike counts. *Under review*.

See Also

bw.joint.dpcirc

Examples

```
data(spikes)
direction<-circular(spikes$direction,units="degrees")
counts<-spikes$counts
output<-kern.dpreg.circ(direction, counts, bw=c(7.41,4.47))

 # Plot the data
plot(as.numeric(output$datax),output$datay,pch=16,xlab="Stimulus direction",
      ylab="Spike number")
# Represent the estimated mean
points(as.numeric(output$x),exp(output$estim[[1]]),type="l",col=2,lwd=2)

# Represent the estimated dispersion
plot(as.numeric(output$x),exp(output$estim[[2]]),type="l",col=2,lwd=2,
      xlab="Stimulus direction",ylab="Dispersion")
```

---

kern.reg.circ.lin *Nonparametric regression estimation for circular data*

Description

Function `kern.reg.circ.lin` implements the Nadaraya-Watson estimator and the Local-Linear estimator for circular-linear data (circular covariate and linear response), as described in Di Marzio et al. (2009) and Oliveira et al. (2013), taking the von Mises distribution as kernel.

Function `kern.reg.circ.circ` implements the Nadaraya-Watson estimator and the Local-Linear estimator for circular-circular data (circular covariate and circular response), as described in Di Marzio et al. (2012), taking the von Mises distribution as kernel.

Function `kern.reg.lin.circ` implements the Nadaraya-Watson estimator and the Local-Linear estimator for linear-circular data (linear covariate and circular response), as described in Di Marzio et al. (2012), taking the Normal distribution as kernel.

Usage

```
kern.reg.circ.lin(x, y, t=NULL, bw, method="LL", from=circular(0),
to=circular(2*pi), len=250, tol=300)
kern.reg.circ.circ(x, y, t=NULL, bw, method="LL", from=circular(0),
to=circular(2*pi), len=250)
kern.reg.lin.circ(x, y, t=NULL, bw, method="LL", len=250)
## S3 method for class 'regression.circular'
print(x, digits=NULL, ...)
```

Arguments

`x`	Vector of data for the independent variable. The object is coerced to class `circular` when using functions `kern.reg.circ.lin` and `kern.reg.circ.circ`.
`y`	Vector of data for the dependent variable. This must be same length as `x`. The object is coerced to class `circular` when using functions `kern.reg.circ.circ` and `kern.reg.lin.circ`.
`t`	Points where the regression function is estimated. If `NULL` equally spaced points are used according to the parameters `from`, `to` and `len`.
`bw`	Smoothing parameter to be used. The value of the smoothing parameter can be chosen by using the function `bw.reg.circ.lin`, `bw.reg.circ.circ` and `bw.reg.lin.circ`.
`method`	Character string giving the estimator to be used. This must be one of `"LL"` for Local-Linear estimator or `"NW"` for Nadaraya-Watson estimator. Default `method="LL"`.
`from, to`	Left and right-most points of the grid at which the regression function is to be estimated. The objects are coerced to class `circular`.
`len`	Number of equally spaced points at which the regression function is to be estimated.
`tol`	Tolerance parameter to avoid overflow when `bw` is larger than `tol`. Default is `tol=300`.
`digits`	Integer indicating the precision to be used.
`...`	further arguments

Details

See Di Marzio et al. (2012). See Section 3 in Oliveira et al. (2013). See Di Marzio et al. (2009). The NAs will be automatically removed.

Value

An object with class "regression.circular" whose underlying structure is a list containing the following components:

`datax, datay`	Original dataset.
`x`	The n coordinates of the points where the regression is estimated.
`y`	The estimated values.
`bw`	The smoothing parameter used.
`N`	The sample size after elimination of missing values.
`call`	The call which produced the result.
`data.name`	The deparsed name of the x argument.
`has.na`	Logical, for compatibility (always FALSE).

Author(s)

Maria Oliveira, Rosa M. Crujeiras and Alberto Rodriguez–Casal

References

Di Marzio, M., Panzera A. and Taylor, C. C. (2009) Local polynomial regression for circular predictors. *Statistics and Probability Letters*, **79**, 2066–2075.

Di Marzio, M., Panzera A. and Taylor, C. C. (2012) Non–parametric regression for circular responses. *Scandinavian Journal of Statistics*, **40**, 228–255.

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2013) Nonparametric circular methods for exploring environmental data. *Environmental and Ecological Statistics*, **20**, 1–17.

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2014) NPCirc: an R package for nonparametric circular methods. *Journal of Statistical Software*, **61**(9), 1–26. https://www.jstatsoft.org/v61/i09/

See Also

`plot.regression.circular`, `lines.regression.circular`

Examples

```
### circular-linear
data(speed.wind2)
dir <- speed.wind2$Direction
vel <- speed.wind2$Speed
nas <- which(is.na(vel))
dir <- circular(dir[-nas],units="degrees")
vel <- vel[-nas]
```

```
estLL <- kern.reg.circ.lin(dir, vel, method="LL")
estNW <- kern.reg.circ.lin(dir, vel, method="NW")
# Circular representation
res<-plot(estNW, plot.type="circle", points.plot=TRUE,
labels=c("N","NE","E","SE","S","SO","O","NO"),
label.pos=seq(0,7*pi/4,by=pi/4), zero=pi/2, clockwise=TRUE)
lines(estLL, plot.type="circle", plot.info=res, line.col=2)
# Linear representation
plot(estNW, plot.type="line", points.plot=TRUE, xlab="direction", ylab="speed (m/s)")
lines(estLL, plot.type="line", line.col=2)


### circular-circular
data(wind)
wind6 <- circular(wind$wind.dir[seq(7,1752,by=24)])
wind12 <- circular(wind$wind.dir[seq(13,1752,by=24)])
estNW <- kern.reg.circ.circ(wind6,wind12,t=NULL,bw=6.1,method="NW")
estLL <- kern.reg.circ.circ(wind6,wind12,t=NULL,bw=2.25,method="LL")
# Torus representation
plot(estNW, plot.type="circle", points.plot=TRUE, line.col=2, lwd=2, points.col=2,
units="degrees")
lines(estLL, plot.type="circle", line.col=3, lwd=2)
# Linear representation
plot(estNW, plot.type="line", points.plot=TRUE, xlab="Wind direction at 6 a.m.",
ylab="Wind direction at noon")
lines(estLL, plot.type="line", line.col=2)


### linear-circular
data(periwinkles)
dist <- periwinkles$distance
dir <- circular(periwinkles$direction, units="degrees")
estNW <- kern.reg.lin.circ(dist,dir,t=NULL,bw=12.7,method="NW")
estLL <- kern.reg.lin.circ(dist,dir,t=NULL,bw=200,method="LL")
# Cylinder representation
plot(estNW, plot.type="circle", points.plot=TRUE, line.col=2, lwd=2, points.col=2)
lines(estLL, plot.type="circle", line.col=3, lwd=2)
# Linear representation
 plot(estNW, plot.type="line", points.plot=TRUE, units="radians", main="")
 lines(estLL, plot.type="line", line.col=2, units="radians")
```

`lines.regression.circular` *Add a plot for circular regression*

Description

The `lines` add a plot for `regression.circular` objects.

Usage

```
## S3 method for class 'regression.circular'
lines(x, plot.type=c("circle", "line"), points.plot=FALSE, rp.type="p", type="l",
line.col=1, points.col="grey", points.pch=1, units=NULL, zero=NULL,
clockwise=NULL, radial.lim=NULL, plot.info=NULL, ...)
```

Arguments

x	An object of class regression.circular.
plot.type	Type of the plot: "line": linear plot, "circle": circular plot.
points.plot	Logical; if TRUE original data are added to the plot.
rp.type, type	Character indicating the type of plotting.
line.col	Color code or name.
points.col	Color code or name for the original data. Used if points.plot=TRUE.
points.pch	Plotting 'character', i.e., symbol to use for the original data. Used if points.plot=TRUE.
units	Units measure used in the plot. If NULL the value is taken from the attribute of object 'x' from the argument 'x', i.e. x$x.
zero	Where to place the starting (zero) point, i.e., the zero of the plot. Ignored if plot.info is provided.
clockwise	Logical, indicating the sense of rotation of the plot: clockwise if TRUE and counterclockwise if FALSE. Ignored if plot.info is provided.
radial.lim	The range of the grid circle. Used if plot.type="circle".
plot.info	An object from plot.regression.circular that contains information on the zero, the clockwise and radial.lim. Used if plot.type="circle".
...	Further arguments to be passed to lines.default (if plot.type="line") or to radial.plot (if plot.type="circle" and x is the output of kern.reg.circ.lin) or to lines3d (if plot.type="circle") and x is the output of kern.reg.circ.circ).

Author(s)

Maria Oliveira, Rosa M. Crujeiras and Alberto Rodriguez–Casal

References

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2014) NPCirc: an R package for nonparametric circular methods. *Journal of Statistical Software*, **61**(9), 1–26. https://www.jstatsoft.org/v61/i09/

See Also

kern.reg.circ.lin, kern.reg.circ.circ, kern.reg.lin.circ, plot.regression.circular

Examples

```
set.seed(1012)
n <- 100
x <- runif(100, 0, 2*pi)
y <- sin(x)+0.5*rnorm(n)
estNW<-kern.reg.circ.lin(circular(x),y,t=NULL,bw=10,method="NW")
estLL<-kern.reg.circ.lin(circular(x),y,t=NULL,bw=10,method="LL")
res<-plot(estNW, plot.type="circle", points.plot=TRUE)
lines(estLL, plot.type="circle",line.col=2, plot.info=res)
```

---

modalreg.circ.lin *Circular multimodal regression estimation*

Description

Function modalreg.circ.lin implements the nonparametric multimodal regression estimator for a circular covariate and a real-valued response, as described in Alonso-Pena and Crujeiras (2022). It takes the von Mises distribution as the kernel associated to the predictor variable and the normal distribution as the kernel associated to the response variable.

Function modalreg.circ.circ implements the nonparametric multimodal regression estimator for a circular covariate and a circular response, as described in Alonso-Pena and Crujeiras (2022). It takes the von Mises distribution as the kernel associated to the predictor variable and the response variables.

Function modalreg.lin.circ implements the nonparametric multimodal regression estimator for a real-valued covariate and a circular response, as described in Alonso-Pena and Crujeiras (2022). It takes the normal distribution as the kernel associated to the predictor variable and the von Mises distribution as the kernel associated to the response variable.

Usage

```
modalreg.circ.lin(x, y, t=NULL, bw=NULL, tol = 0.0001, maxit = 500,
    from = circular(0),to = circular(2 * pi), len = 300)

modalreg.circ.circ(x, y, t=NULL, bw=NULL, tol = 0.00001,
    maxit = 500, from = circular(0), to = circular(2 * pi), len = 300)

modalreg.lin.circ(x, y, t=NULL, bw=NULL, tol = 0.0001, maxit = 500, len=300)
```

Arguments

x	Vector of data for the independent variable. The object is coerced to class circular when using functions modalreg.circ.lin and modalreg.circ.circ.
y	Vector of data for the dependent variable. This must be same length as x. The object is coerced to class circular when using functions modalreg.circ.circ and modalreg.lin.circ.

t	Points where the regression function is estimated. If NULL, equally spaced points are used according to the parameters from, to and len.
bw	Vector of length two with the values of the smoothing parameters to be used. The first component corresponds to the smoothing parameter associated to the predictor variable and the second component is the parameter associated to the response variable. If NULL, the parameters are selected via modal cross-validation.
tol	Tolerance parameter for convergence in the estimation through the conditional (circular) mean shift.
maxit	Maximum number of iterations in the estimation through the conditional (circular) mean shift.
from, to	Left and right-most points of the grid at which the regression multifunction is to be estimated. The objects are coerced to class circular.
len	Number of equally spaced points at which the regression multifunction is to be estimated.

Details

See Alonso-Pena and Crujeiras (2022) for details.

The NAs will be automatically removed.

Value

A list containing the following components:

datax, datay	Original dataset.
x	The n coordinates of the points where the regression multifunction is estimated.
y	A list with dimension the length of the number of evaluation points containing the estimated values of the multidunfunction for each evaluation point.
bw	A verctor of lenght two with the smoothing parameters used.
n	The sample size after elimination of missing values.
call	The call which produced the result.
data.name	The deparsed name of the x argument.
has.na	Logical, for compatibility (always FALSE).

Author(s)

Maria Alonso-Pena and Rosa M. Crujeiras.

References

Alonso-Pena, M. and Crujeiras, R. M. (2022). Analizing animal escape data with circular nonparametric multimodal regression. Annals of Applied Statistics. (To appear).

See Also

bw.modalreg.circ.lin, bw.modalreg.circ.circ, bw.modalreg.lin.circ

Examples

```
# Circ-lin
set.seed(8833)
n1<-100
n2<-100
gamma<-8
sigma<-1.5
theta1<-rcircularuniform(n1)
theta2<-rcircularuniform(n2)
theta<-c(theta1,theta2)
y1<-2*sin(2*theta1)+rnorm(n1,sd=sigma)
y2<-gamma+2*sin(2*theta2)+rnorm(n2,sd=sigma)
y<-as.numeric(c(y1,y2))
fit<-modalreg.circ.lin(theta,y,bw=c(10,1.3))

# Lin-circ
n1<-100
n2<-100
con<-8
set.seed(8833)
x1<-runif(n1)
x2<-runif(n2)
phi1<-(6*atan(2.5*x1-3)+rvonmises(n1,m=0,k=con))
phi2<-(pi+6*atan(2.5*x2-3)+rvonmises(n2,m=0,k=con))
x<-c(x1,x2)
phi<-c(phi1,phi2)
fit<-modalreg.lin.circ(x, phi, bw=c(0.1,2.5))

# Circ-circ
n1<-100
n2<-100
con<-10
set.seed(8833)
theta1<-rcircularuniform(n1)
theta2<-rcircularuniform(n2)
phi1<-(2*cos(theta1)+rvonmises(n1,m=0,k=con))
phi2<-(3*pi/4+2*cos(theta2)+rvonmises(n2,m=0,k=con))
theta=c(theta1,theta2)
phi=c(phi1,phi2)
fit<-modalreg.circ.circ(theta, phi, bw=c(30,3))
```

`noeffect.circ.lin` *No-effect test for regression with circular data*

Description

Function `noeffect.circ.lin` computes the no-effect test for a circular predictor variable and a real-valued response variable as described in Alonso-Pena et al. (2021). It uses the nonparametric

Nadaraya-Watson estimator or the Local-Linear estimator for circular-linear data described in Di Marzio et al. (2009) and Oliveira et al. (2013).

Function `noeffect.lin.circ` computes the no-effect test for a real-valued predictor variable and a circular response variable as described in Alonso-Pena et al. (2021). It uses the nonparametric Nadaraya-Watson estimator or the Local-Linear estimator for linear-circular data described in Di Marzio et al. (2012).

Function `noeffect.circ.circ` computes the no-effect test for a circular predictor variable and a circular response variable as described in Alonso-Pena et al. (2021). It uses the nonparametric Nadaraya-Watson estimator or the Local-Linear estimator for circular-circular data described in Di Marzio et al. (2012).

Usage

```
noeffect.circ.lin(x, y, bw, method = "LL", calib = "chisq", n_boot = 500)
noeffect.lin.circ(x, y, bw, method = "LL", n_boot = 500)
noeffect.circ.circ(x, y, bw, method = "LL", n_boot = 500)
```

Arguments

`x`	Vector of data for the independent variable. The object is coerced to class `circular` when using functions `noeffect.circ.lin` and `noeffect.circ.circ`.
`y`	Vector of data for the dependent variable. This must be same length as `x`. The object is coerced to class `circular` when using functions `noeffect.lin.circ` and `noeffect.circ.circ`.
`bw`	Smoothing parameter to be used. If not provided, functions `noeffect.circ.lin` and `noeffect.circ.circ` select $4cv$ and function `noeffect.lin.circ` selects $cv/4$, where cv is the parameter selected by cross-validation.
`method`	Character string giving the estimator to be used. This must be one of `"LL"` for Local-Linear estimator or `"NW"` for Nadaraya-Watson estimator. Default `method="LL"`.
`calib`	Character string giving the calibration method to be used in `noeffect.circ.lin` function. This must be one of `"chisq"` for the chi-squared approximation or `"boot"` for the bootstrap calibration.
`n_boot`	Number of bootstrap resamples. Default is `n_boot=500`. In function `noeffect.circ.lin`, only if `calib="boot"`.

Details

See Alonso-Pena et al. (2021). The NAs will be automatically removed.

Value

A list with class `"htest"` containing the following components:

`statistic`	observed value of the statistic.
`bw`	Smoothing parameter used.
`p.value`	p-value for the test.

`data.name`	a character string giving the name(s) of the data.
`alternative`	a character string describing the alternative hypothesis.

Author(s)

Maria Alonso-Pena, Jose Ameijeiras-Alonso and Rosa M. Crujeiras

References

Alonso-Pena, M., Ameijeiras-Alonso, J. and Crujeiras, R.M. (2021) Nonparametric tests for circular regression. *Journal of Statistical Computation and Simulation*, **91**(3), 477–500.

Di Marzio, M., Panzera A. and Taylor, C. C. (2009) Local polynomial regression for circular predictors. *Statistics and Probability Letters*, **79**, 2066–2075.

Di Marzio, M., Panzera A. and Taylor, C. C. (2012) Non–parametric regression for circular responses. *Scandinavian Journal of Statistics*, **40**, 228–255.

Oliveira, M., Crujeiras R.M. and Rodriguez-Casal, A. (2013) Nonparametric circular methods for exploring environmental data. *Environmental and Ecological Statistics*, **20**, 1–17.

See Also

`kern.reg.circ.lin`, `kern.reg.lin.circ`, `kern.reg.circ.circ`

Examples

```
# No-effect circ-lin
set.seed(2025)
x <- rcircularuniform(200)
y <- 2*sin(as.numeric(x)) + rnorm(200, sd=2)
noeffect.circ.lin(x, y)

# No-effect lin-circ
set.seed(2025)
x <- runif(200)
y<- pi/8 + rvonmises(200, mu = 0, kappa = 0.75)
noeffect.lin.circ(x, y)


# No-effect circ-circ
set.seed(2025)
x <- rcircularuniform(200)
y <- atan2(sin(2*x),cos(2*x)) + rvonmises(200, mu = 0, kappa = 2)
noeffect.circ.circ(x, y)
```

periwinkles *Orientations of dragonflies*

Description

These data, presented in Fisher and Lee (1992), contain distance and directions of movements from small blue periwinkles after relocation.

Usage

```
data(periwinkles)
```

Format

A two-column data frame with 73 observations. Distances are measured in centimeters and directions are measured in degrees.

Source

Fisher, N. I. and Lee, A. J. (1992) Regression models for angular responses. *Biometrics*, **48**, 665–677.

Examples

```
data(periwinkles)
```

plot.regression.circular *Plot circular regression*

Description

The `plot` method for `regression.circular` objects.

Usage

```
## S3 method for class 'regression.circular'
plot(x, plot.type=c("circle", "line"), points.plot=FALSE, rp.type="p", type="l",
line.col=1, points.col="grey", points.pch=1, xlim=NULL, ylim=NULL,
radial.lim=NULL, xlab=NULL, ylab=NULL, labels=NULL, label.pos=NULL, units=NULL,
zero=NULL, clockwise=NULL, main=NULL, ...)
```

Arguments

`x`	An object of class `regression.circular`.
`plot.type`	Type of the plot: "line": linear plot, "circle": circular plot.
`points.plot`	Logical; if TRUE original data are added to the plot.
`rp.type, type`	Character indicating the type of plotting. Default `type="l"` and `rp.type="p"`.
`line.col`	Color code or name.
`points.col`	Color code or name for the original data. Used if `points.plot=TRUE`.
`points.pch`	Plotting 'character', i.e., symbol to use for the original data. Used if `points.plot=TRUE`.
`xlim, ylim`	The ranges to be encompassed by the x and y axes. Used if `plot.type="line"`.
`radial.lim`	The range of the grid circle, used if `plot.type="circle"`.
`xlab, ylab`	Titles for the x axis and y axis, respectively.
`labels`	Character or expression vector of labels to be placed at the `label.pos`. `label.pos` must also be supplied.
`label.pos`	Vector indicating the position (between 0 and 2π) at which the labels are to be drawn.
`units`	Units measure used in the plot. If `NULL` the value is taken from the attribute of object 'x' from the argument 'x', i.e. x$x.
`zero`	Where to place the starting (zero) point, i.e. the zero of the plot. If `NULL` the value is taken from the attribute of object 'x' from the argument 'x', i.e. x$x
`clockwise`	Logical, indicating the sense of rotation of the plot: clockwise if `TRUE` and counterclockwise if `FALSE`. If `NULL` the value is taken from the attribute of object 'x' from the argument 'x', i.e. x$x
`main`	An overall title for the plot.
`...`	Further arguments to be passed to `plot.default` (if `plot.type="line"`) or to to `radial.plot` (if codeplot.type="circle" and `x` is the output of `kern.reg.circ.lin`) or to `lines3d` (if `plot.type="circle"`) and `x` is the output of `kern.reg.circ.circ`).

Value

If `plot.type="circle"` and `x` is the output of `kern.reg.circ.lin`), this function returns a list with information on the plot: `zero`, `clockwise` and `radial.lim`.

Author(s)

Maria Oliveira, Rosa M. Crujeiras and Alberto Rodriguez–Casal

References

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2014) NPCirc: an R package for nonparametric circular methods. *Journal of Statistical Software*, **61**(9), 1–26. `https://www.jstatsoft.org/v61/i09/`

See Also

`kern.reg.circ.lin`, `kern.reg.circ.circ`, `kern.reg.lin.circ`, `lines.regression.circular`

Examples

```
set.seed(1012)
n <- 100
x <- runif(100, 0, 2*pi)
y <- sin(x)+0.5*rnorm(n)
estNW<-kern.reg.circ.lin(circular(x),y,t=NULL,bw=10,method="NW")
plot(estNW, plot.type="line", points.plot=TRUE)
plot(estNW, plot.type="circle", points.plot=TRUE)
```

---

pm10 *Pm10 particles in Pontevedra, Spain*

Description

This dataset contains measurements on pm10 particles concentration in the city of Pontevedra, Spain, as well as wind speed and wind direction recordings. The measurements were taken in a meteorological station in Pontevedra, during the year 2019, every six hours.

Usage

```
data("pm10")
```

Format

A data frame with 1156 observations on the following 3 variables.

- `pm10` A numeric vector containing the concentration of pm10 particles (in micrograms per cubic meter).
- `direction` A numeric vector containing the wind direction in degrees between 0 and 360.
- `speed` A numeric vector containing the wind speed measured in kilometers per hour.

Source

Meteogalicia https://www.meteogalicia.gal/.

Examples

```
data(pm10)
```

`sandhoppers` *Behavioral plasticity of Talitrus saltator and Talorchestia brito*

Description

Orientation measured under natural conditions and other variables of interest for analyzing the behavioral plasticity of two sympatric sandhoppers species, Talitrus saltator and Talorchestia brito. The experiment was carried out on the exposed nontidal sand of Zouara beach located in the Tunisian northwestern coast. More details can be found in Marchetti and Scapini (2003) or Scapini et al. (2002).

Usage

```
data("sandhoppers")
```

Format

A data frame with 1828 observations on the following 12 variables.

`angle` Numeric vector containing the orientation angles in radians between 0 and 2π.

`date` A factor where each level indicates the date when angles were measured.

`month` A factor with two levels indicating the month when angles were measured. Experiments were performed in two different periods, `April` and `October`, which were chosen for the abundance of the populations, as well as for their non-extreme and changing climatic conditions.

`time` A factor with levels `afternoon`, `morning` and `noon`.

`azim` A numeric vector indicating the sun azimuth. The sun position was confounded with the time of the day (morning: 100-150, noon: az=151-210 and afternoon: az=211-260 experiments).

`hour` A factor with hours when angles were measured.

`species` A factor with three levels (`brito`, `salt`, `ND`) indicating the specie (brito, saltator, not determined).

`sex` A factor with three levels (`F`, `M`, `J`) indicating the sex (female, male, J).

`temp` A numeric vector indicating the temperature (degrees centigrade).

`humid` A numeric vector indicating the air relative humidity (%).

`land` A factor with two levels (`no`, `yes`) indicating landscape view was either permitted or screened.

`trap` A numeric vector containing the traps identifier used for capturing the sandhoppers.

Details

See Marchetti and Scapini (2003) and Scapini et al. (2002).

Source

Authors thank Prof. Felicita Scapini for providing the sandhoppers data (collected under the support of the European Project ERB ICI8-CT98-0270).

References

Marchetti, G. M. and Scapini, F. (2003). Use of multiple regression models in the study of sandhopper orientation under natural conditions. *Estuarine, Coastal and Shelf Science*, **58**, 207-215.

Scapini, F., Aloia, A., Bouslama, M. F., Chelazzi, L., Colombini, I., ElGtari, M., Fallaci, M. and Marchetti, G. M. (2002). Multiple regression analysis of the sources of variation in orientation of two sympatric sandhoppers, Talitrus saltator and Talorchestia brito, from an exposed Mediterranean beach. *Behavioral Ecology and Sociobiology*, **51**(5), 403–414.

Examples

```
data(sandhoppers)
```

`speed.wind` *Wind speed and wind direction data*

Description

This dataset consists of hourly observations of wind direction and wind speed in winter season (from November to February) from 2003 until 2012 in the Atlantic coast of Galicia (NW–Spain). Data are registered by a buoy located at longitude -0.210E and latitude 43.500N in the Atlantic Ocean. The dataset `speed.wind2`, analized in Oliveira et al. (2013), is a subset of `speed.wind` which is obtained by taking the observations with a lag period of 95 hours.

Usage

```
data(speed.wind)
data(speed.wind2)
```

Format

`speed.wind` is a data frame with 19488 observations on six variables: day, month, year, hour, wind speed (in m/s) and wind direction (in degrees). `speed.wind2` is a subset with 200 observations.

Details

Data contains NAs. There is no data in November 2005, December 2005, January 2006, February 2006, February 2007, February 2009 and November 2009. Months of November 2004, December 2004, January 2007, December 2009 are not fully observed.

Source

Data can be freely downloaded from the Spanish Portuary Authority (https://www.puertos.es/).

References

Oliveira, M., Crujeiras, R.M. and Rodriguez–Casal (2014) CircSiZer: an exploratory tool for circular data. *Environmental and Ecological Statistics*, **21**, 143–159.

Examples

```
data(speed.wind2)

# Density
dir <- circular(speed.wind2$Direction, units="degrees", template="geographics")
plot(dir, stack=TRUE,  shrink= 1.1)
rose.diag(dir, bins=16, add=TRUE)
lines(kern.den.circ(dir,bw=1), lwd=2, col=2)
lines(kern.den.circ(dir,bw=10), lwd=2, col=3)
lines(kern.den.circ(dir,bw=40), lwd=2, col=4)

# Regression
vel <- speed.wind2$Speed
nas <- which(is.na(vel))
dir <- dir[-nas]
vel <- vel[-nas]
res<-plot(kern.reg.circ.lin(dir, vel, bw=1, method="LL"), plot.type="circle",
points.plot=TRUE, line.col=2, lwd=2, main="")
lines(kern.reg.circ.lin(dir, vel, bw=10, method="LL"), plot.type="circle", plot.info=res,
line.col=3, lwd=2)
lines(kern.reg.circ.lin(dir, vel, bw=40, method="LL"), plot.type="circle", plot.info=res,
line.col=4, lwd=2)
```

---

spikes *Neuronal spikes in a macaque monkey*

Description

This dataset, given by Kohn and Movshon (2003), contains observations from an experimental study on macaque monkeys, where a macaque received a visual stimulus. The dataset is composed of 68 observations, consisting on the angle of stimulus and the number of spikes produced in a MT/V5 neuron.

Usage

```
data("spikes")
```

Format

A data frame with 68 observations on the following 2 variables.

`direction` A numeric vector containing the direction of the visual stimulus (in degrees).

`counts` A numeric vector containing the number of spikes.

Details

See Kohn and Movshon (2003) for details. This dataset was also studied in Alonso-Pena et al (2022).

Source

The dataset was obtained by Kohn and Movshon (2003).

References

Kohn, A., and Movshon, J.A. (2003). Neuronal adaptation to visual motion in area MT of the macaque. *Neuron*, **39**(4), 681–691.

Alonso-Pena, M., Gijbels, I. and Crujeiras, R.M. (2022). Flexible joint modeling of mean and dispersion for the directional tuning of neuronal spike counts. *Under review*.

Examples

```
data(spikes)
```

---

temp.wind *Temperature and wind direction data*

Description

These data, analyzed by Oliveira et al. (2013), consists of observations of temperature and wind direction during the austral summer season 2008-2009 (from November 2008 to March 2009) in Vinciguerra (Tierra del Fuego, Argentina).

Usage

```
data(temp.wind)
```

Format

A data frame with 3648 observations on four variables: `Date`, `Time`, `Temperature` (in degrees Celsius) and `Direction` (in degrees).

Details

Data contains NAs.

Source

The authors want to acknowledge Prof. Augusto P?rez–Alberti for providing the data, collected within the Project POL2006-09071 from the Spanish Ministry of Education and Science.

References

Oliveira, M., Crujeiras R.M. and Rodriguez–Casal, A. (2013) Nonparametric circular methods for exploring environmental data. *Environmental and Ecological Statistics*, **20**, 1–17.

Examples

```
data(temp.wind)
winddir <- temp.wind[,4]
temp <- temp.wind[,3]
nas <- which(is.na(winddir))
winddir <- circular(winddir[-nas], units="degrees")
temp <- temp[-nas]

est <- kern.reg.circ.lin(winddir, temp, t=NULL, bw=3.41, method="LL")
plot(est, plot.type="line", xlab="wind direction", ylab="temperature")
plot(est, plot.type="circle", points.plot=TRUE)
```

---

wind — *Wind direction data*

Description

This dataset consists of hourly observations of wind direction measured at a weather station in Texas from May 20 to July 31, 2003 inclusive.

Usage

```
data(wind)
```

Format

`wind` is a data frame with observations on three variables: data, hour and wind direction (in radians).

Source

The data, which correspons to the weather station designated as C28_1, are part of a larger data set taken from the Codiac data archive, available at https://data.eol.ucar.edu/dataset/85.034. The full data set contains hourly resolution surface meteorological data from the Texas Natural Resources Conservation Commission Air Quality Monitoring Network, from May 20 to July 31, 2003 inclusive. These data are provided by NCAR/EOL under the sponsorship of the National Science Foundation.

References

Kato, S. and Jones, M. C. (2010) A family of distributions on the circle with links to, and applications arising from, M?bius transformation. *Journal of the American Statistical Association*, **105**, 249–262.

Di Marzio, M., Panzera A. and Taylor, C. C. (2012) Non–parametric regression for circular responses. *Scandinavian Journal of Statistics*, **40**, 228–255.

Examples

```
data(wind)
```

`zebrafish` *Zebrafish*

Description

The data consists on measurements from an experimental study on larval zebrafish, which were startled by an imitating predator consisting on a robot disguised as an adult zebrafish. The dataset includes 502 observations corresponding to the escape directions of each fish and the angles in which the zebrafish perceive the threat.

Usage

```
data("zebrafish")
```

Format

A data frame with 502 observations on the following 2 variables.

`stimulus` A numeric vector containing the angles of stimulus (in radians)

`res_angle` A numeric vector containing the directions of escape (in radians)

Details

Analysis of the zebrafish data with circular regression methods can be seen in Alonso-Pena et al. (2022).

Source

The data were obtained from the Dryad Digital Repository doi:10.5061/dryad.47mq9 and first analyzed in Nair et al. (2017).

References

Nair, A., Changsing, K., Stewart, W.J. and McHenry, M.J. (2017). Fish prey change strategy with the direction of a threat. *Proceedings of the Royal Society B: Biological Sciences*, **284**, 20170393.

Alonso-Pena, M. and Crujeiras, R. M. (2022). Analizing animal escape data with circular nonparametric multimodal regression. Annals of Applied Statistics. (To appear).

Examples

```
data(zebrafish)
```

Index