

Package ‘HRW’

January 23, 2026

Version 1.0-6

Date 2026-01-23

Title Datasets, Functions and Scripts for Semiparametric Regression
Supporting Harezlak, Ruppert & Wand (2018)

Maintainer Matt P. Wand <matt.wand@uts.edu.au>

Imports KernSmooth, grDevices, graphics, splines, stats

Suggests fields, lattice, mgcv, Ecdat, LowRankQP, VGAM, e1071, fda,
fda.usc, mlbench, nlme, quantreg, refund, rgl, rstan

Description The book ``Semiparametric Regression with R'' by J. Harezlak, D. Ruppert & M.P. Wand (2018, Springer; ISBN: 978-1-4939-8851-8) makes use of datasets and scripts to explain semiparametric regression concepts. Each of the book's scripts are contained in this package as well as datasets that are not within other R packages. Functions that aid semiparametric regression analysis are also included.

License GPL (>= 2)

LazyData TRUE

LazyDataCompression xz

NeedsCompilation no

Depends R (>= 4.5.0)

Author Jaroslaw Harezlak [aut],
David Ruppert [aut],
Matt P. Wand [aut, cre]

Repository CRAN

Date/Publication 2026-01-23 06:41:29 UTC

Contents

BanglaContrac	2
BCR	3
BostonMortgages	4
brainImage	5

capm	6
carAuction	7
CHD	9
coral	10
createBoundary	11
femSBMD	12
growthIndiana	13
indonRespir	14
lidar	15
ozoneSub	16
plankton	17
pointsInPoly	18
protein	19
ragweed	20
scallop	21
schoolResults	22
summMCMC	23
SydneyRealEstate	24
SydneyRealEstateBdry	26
TreasuryRate	27
UtahPEF	28
WarsawApts	29
yields	30
ZOSull	33
Index	35

Description

The BanglaContrac data frame has multilevel data from the 1988 Bangladesh Fertility Survey. There are data on contraceptive use, number of children, age and urban-dweller status of 1,934 women grouped in 60 districts.

Usage

```
data(BanglaContrac)
```

Format

This data frame contains the following columns:

`districtID` district identification number.

`usingContraception` indicator that woman is using contraception:

1 = woman using contraception at time of survey,

0 = woman not using contraception at time of survey.

`childCode` numerical code for number of living children:

- 1 = no living children at time of survey,
- 2 = one living child at time of survey,
- 3 = two living children at time of survey,
- 4 = three or more living children at time of survey.

`ageMinusMean` age in years of woman at time of survey, with mean age subtracted

`isUrban` indicator that woman lives in an urban region:

- 1 = woman an urban region dweller at time of survey,
- 0 = woman a rural region dweller at time of survey.

Source

Huq, N.M. and Cleland, J. (1990). *Bangladesh Fertility Survey 1989 (Main Report)*. Dhaka, Bangladesh: National Institute of Population Research and Training.

Examples

```
library(HRW) ; data(BanglaContrac)
if (require("lattice"))
  print(xyplot(jitter(usingContraception) ~ ageMinusMean | factor(districtID),
    groups = childCode, data = BanglaContrac))
```

BCR

Berry, Carroll and Ruppert (2002) data

Description

The BCR data frame has data from a 6-week clinical trial of a drug versus a placebo. The data are subject to measurement error and have been transformed and rescaled from their original form. These data are analyzed in the 2002 ‘Journal of the American Statistical Association’ article by Berry, Carroll and Ruppert (full reference below).

Usage

```
data(BCR)
```

Format

This data frame contains the following columns:

`status` code for status of patient:

- control = patient is in placebo group,
- treatment = patient is in treatment group.

`w` physician-assessed score of patient’s mental health at baseline.

`y` physician-assessed score of patient’s mental health at end of the study.

References

Berry, S.M., Carroll, R.J. and Ruppert, D. (2002). Bayesian smoothing and regression splines for measurement error problems. *Journal of the American Statistical Association*, **97**, 160-169.

Examples

```
library(HRW) ; data(BCR)
if (require("lattice"))
  print(xypplot(y ~ w|status,data = BCR))
```

BostonMortgages

Mortgage applications in Boston

Description

The full dataset is in the data frame ‘Hdma’ within the R package Ecdat. This data frame is the subset of mortgage applications during the years 1998-1999.

Usage

```
data(BostonMortgages)
```

Format

A data frame with 2,380 observations on the following 13 variables:

`dir` ratio of the debt payments to the total income.
`hir` ratio of the housing expenses to the total income.
`lvr` ratio of the loan size to the assessed value of property.
`ccs` a credit score ranging from 1 to 6, where a low value indicates low credit risk.
`mcs` a mortgage credit score from 1 to 4, where a low value indicates low credit risk.
`pbcr` did the applicant have a public bad credit record?: a factor with levels no and yes.
`dmi` was the applicant denied mortgage insurance?: a factor with levels no and yes.
`self` was the applicant self-employed?: a factor with levels no and yes.
`single` is the applicant single?: a factor with levels no and yes.
`uria` 1989 Massachusetts unemployment rate in the applicant’s industry.
`condominium` is the unit a condominium?: a factor with levels no and yes.
`black` is the applicant black?: a factor with levels no and yes.
`deny` was the mortgage denied?: a factor with levels no and yes.

References

Munnell, A. H., Tootell, G. M. B., Browne, L. E., and McEneaney, J. (1996). Mortgage lending in Boston: Interpreting HDMA data, *American Economic Review*, 25-53.

Examples

```
library(HRW) ; data(BostonMortgages)
BostonMortgages$denyBinary <- as.numeric(BostonMortgages$deny == "yes")
fit <- glm(denyBinary ~ black + dir + lvr + pbcr + self + single + as.factor(ccs),
           family = binomial, data = BostonMortgages)
summary(fit)
```

brainImage

Brain image

Description

The `brainImage` data frame corresponds a functional magnetic image of a coronal slice of a human brain. The data are brain activity on a pixel array.

Usage

```
data(brainImage)
```

Format

This data frame is a 80 by 37 array. The entries correspond to brain activity in each of the corresponding pixels. The columns names are c1-c37. These names have no meaning, and are present to ensure that this data frame conforms with R data frame conventions.

Source

Landman, B.A., Huang, A.J., Gifford, A., Vikram, D.S., Lim, I.A.L., Farrell, J.A.D., Bogovic, J.A., Hua, J., Chen, M., Jarso, S., Smith, S.A., Joel, S., Mori, S., Pekar, J.J., Barker, P.B., Prince, J.L. and van Zijl, P.C.M. (2010). Multi-parametric neuroimaging reproducibility: A 3T resource study. *NeuroImage*, **54**, 2854-2866.

Examples

```
library(HRW) ; data(brainImage)
image(as.matrix(brainImage))
```

capm*Stock indices*

Description

Daily returns on the Standard & Poors' 500 stock market index, daily rate of the U.S. Treasury bills, and 3 companies' stocks including Microsoft, the General Electric and the Ford Motor Company during the period from November 1, 1993 to March 31, 2003.

Usage

```
data(capm)
```

Format

A data frame with 2363 observations on the following 6 variables:

`Close.tbill` Daily Treasury bill rate expressed as a percentage.
`Close.msft` Daily closing price of the Microsoft stock.
`Close.sp500` Daily closing Standard and Poor's 500 index.
`Close.ge` Daily closing price of the General Electric stock.
`Close.ford` Daily closing price of the Ford Motor Company stock.
`Date` Dates from November 1, 1993 to March 31, 2003 (d-Mon-yy and dd-Mon-yr formats).

Source

Federal Reserve Bank of St. Louis U.S.A. (Treasury bill rates) and Yahoo Finance (stock prices).

Examples

```
# The Capital Asset Pricing Model (CAPM) states that the excess returns on a stock
# have a linear relationship with the returns on the market. This example investigates
# the CAPM for General Electric stock:
```

```
library(HRW) ; data(capm)
n <- dim(capm)[1]
riskfree <- capm$Close.tbill[2:n]/365
elrGE <- diff(log(capm$Close.ge)) - riskfree
elrSP500 <- diff(log(capm$Close.sp500)) - riskfree
plot(elrSP500,elrGE,col = "blue",cex = 0.2)
fitOLS <- lm(elrGE ~ elrSP500)
summary(fitOLS)
par(mfrow = c(2,2)) ; plot(fitOLS)
```

carAuction	<i>Cars purchased at auction</i>
------------	----------------------------------

Description

The carAuction data frame has data on several variables concerning cars purchased at automobile auctions by automobile dealerships in the United States of America. The origin of these data is a classification competition titled “Don’t Get Kicked!” that ran on the ‘kaggle’ platform (<https://www.kaggle.com>) during 2011-2012. Many of the variables in this data frame have been derived from those in the original data set from <https://www.kaggle.com>.

Usage

```
data(carAuction)
```

Format

This data frame contains the following columns:

`RefId` unique number assigned to each vehicles.

`IsBadBuy` indicator that the vehicle purchased at auction by an automobile dealership has serious problems that hinder or prevent it being sold - a "bad buy":

1 = the vehicle is a "bad buy"

0 = the vehicle is a "good buy".

All other indicator variables are defined in this way.

`purchIn2010` indicator that vehicle was purchased in 2010.

`aucEqAdesa` indicator that the auction provider at which the vehicle was purchased was Adesa.

`aucEqManheim` indicator that the auction provider at which the vehicle was purchased was Manheim.

`vehYearEq03` indicator that the manufacturer’s year of the vehicle is 2003.

`vehYearEq04` indicator that the manufacturer’s year of the vehicle is 2004.

`vehYearEq05` indicator that the manufacturer’s year of the vehicle is 2005.

`vehYearEq06` indicator that the manufacturer’s year of the vehicle is 2006.

`vehYearEq07` indicator that the manufacturer’s year of the vehicle is 2007.

`ageAtSale` age of the vehicle in years when sold.

`makeEqChevrolet` indicator that the vehicle’s manufacturer is Chevrolet.

`makeEqFord` indicator that the vehicle’s manufacturer is Ford.

`makeEqDodge` indicator that the vehicle’s manufacturer is Dodge.

`makeEqChrysler` indicator that the vehicle’s manufacturer is Chrysler.

`trimEqBas` indicator that the trim level of the vehicle is ‘Bas’.

`trimEqLS` indicator that the trim level of the vehicle is ‘LS’.

`trimEqSE` indicator that the trim level of the vehicle is ‘SE’.

subModelEq4DSEDAN indicator that the submodel of the vehicle is '4DSedan'.
 subModelEq4DSEDANLS indicator that the submodel of the vehicle is '4DSedanLS'.
 subModelEq4DSEDANSE indicator that the submodel of the vehicle is '4DSedanSE'.
 colourEqSilver indicator that the vehicle color is silver.
 colourEqWhite indicator that the vehicle color is white.
 colourEqBlue indicator that the vehicle color is blue.
 colourEqGrey indicator that the vehicle color is grey.
 colourEqBlack indicator that the vehicle color is black.
 colourEqRed indicator that the vehicle color is red.
 colourEqGold indicator that the vehicle color is gold.
 colourEqOrange indicator that the vehicle color is orange.
 transEqManual indicator that the vehicle has manual transmission.
 wheelEqAlloy indicator that the vehicle has alloy wheels.
 wheelEqCovers indicator that the vehicle has covered wheels.
 odomRead the vehicle's odometer reading in miles.
 AmericanMade indicator that the vehicle was manufactured in the United States of America.
 otherAsianMade indicator that the vehicle was manufactured in an Asian nation other than Japan or South Korea.
 sizeEqTruck indicator that the size category of the vehicle is 'truck'.
 sizeEqMedium indicator that the size category of the vehicle is 'medium'.
 sizeEqSUV indicator that the size category of the vehicle is 'SUV'.
 sizeEqCompact indicator that the size category of the vehicle is 'compact'.
 sizeEqVan indicator that the size category of the vehicle is 'van'.
 price acquisition price for this vehicle in average condition at time of purchase in U.S. dollars.
 purchInTexas indicator that the vehicle was purchased in Texas.
 purchInFlorida indicator that the vehicle was purchased in Florida.
 purchInCalifornia indicator that the vehicle was purchased in California.
 purchInNorthCarolina indicator that the vehicle was purchased in North Carolina.
 purchInArizona indicator that the vehicle was purchased in Arizona.
 purchInColorado indicator that the vehicle was purchased in Colorado.
 purchInSouthCarolina indicator that the vehicle was purchased in South Carolina.
 costAtPurch acquisition cost paid for the vehicle at time of purchase.
 onlineSale indicator that the vehicle was purchased online.
 warrantyCost warranty cost in U.S. dollars.

Source

The "Don't Get Kicked" competition, <https://www.kaggle.com>.

Examples

```
library(HRW) ; data(carAuction)
## Not run:
for (colNum in 3:10)
{
  plot(jitter(carAuction[,colNum]),jitter(carAuction$IsBadBuy),pch = ".",
       xlab = names(carAuction)[colNum],ylab = "is car a bad buy?",col = "blue")
  readline("Hit Enter to continue.\n")
}
for (colNum in 11:51)
{
  plot(jitter(carAuction[,colNum]),jitter(carAuction$IsBadBuy),pch = ".",
       xlab = names(carAuction)[colNum],ylab = "is car a bad buy?",col = "blue")
  readline("Hit Enter to continue.\n")
}

## End(Not run)
```

CHD

Coronary heart disease

Description

The CHD data frame has data on coronary heart disease status, cholesterol level measurements and age. Further details are given in the 1996 ‘Journal of the American Statistical Association’ article by Roeder, Carroll and Lindsay (full reference below).

Usage

```
data(CHD)
```

Format

This data frame contains the following columns:

CHD indicator of coronary heart disease status:
 0 = patient does not have coronary heart disease,
 1 = patient has coronary heart disease.

LDL low density lipoprotein cholesterol level.

TC total cholesterol level.

age age of patient in years.

Source

Roeder, K., Carroll, R.J. and Lindsay, B.G. (1996). A semiparametric mixture approach to case-control studies with errors in covariates. *Journal of the American Statistical Association*, **91**, 722-732.

References

Richardson, S., Leblond, L., Jausset, I. and Green, P.J. (2002). Mixture models in measurement error problems, with reference to epidemiological studies. *Journal of the Royal Statistical Society, Series A*, **163**, 549-566.

Examples

```
library(HRW) ; data(CHD)
pairs(CHD)
```

coral

Coral organisms in French Polynesia

Description

The coral data frame has data on initial size, taxonomic identity and alive/death status of coral organisms in French Polynesia.

Usage

```
data(coral)
```

Format

This data frame contains the following columns:

siteDepthPeriod factor with levels corresponding to a code for the site, depth and time period concerning where and when coral organisms were measured.

taxon factor corresponding to an abbreviation for taxonomic identity:

ACR = Acropora,

POC = Pocillopora,

POR = Porites.

logInitialSizePlusOne initial size measurement of coral organism transformed according to the $\log(\text{initial size} + 1)$.

died indicator that coral organism has died:

1 = coral organism has died,

0 = coral organism still alive.

Source

Kayal, M., Vercelloni, J., Wand, M.P. and Adjeroud, M. (2015). Searching for the best bet in life-strategy: a quantitative population dynamics approach to life history trade-offs in reef-building corals. *Ecological Complexity*, **23**, 73-84.

Examples

```
library(HRW) ; data(coral)
if (require("lattice"))
  print(xyplot(died ~ logInitialSizePlusOne|siteDepthPeriod*taxon,
               data = coral, layout = c(15,5)))
```

createBoundary	<i>Boundary polygon creation</i>
----------------	----------------------------------

Description

Create a boundary polygon corresponding nominally to the effective probability density support of a bivariate dataset via an interactive graphical interface and mouse (or, possibly, touchpad) positionings and button clicks.

Usage

```
createBoundary(x, y)
```

Arguments

x	vector containing the x-coordinates of a bivariate dataset.
y	vector containing the y-coordinates of a bivariate dataset.

Details

After the bivariate dataset is displayed on the screen a boundary polygon is selected by performing left mouse (or, possibly, touchpad) clicks on the screen to specify vertex positions, and then moving around in a clockwise direction until the polygon is completed. Completion is achieved by clicking inside the red octagon surrounding the starting vertex.

Value

A two-column matrix containing the vertices of the selected boundary polygon.

Author(s)

M.P. Wand <matt.wand@uts.edu.au>

See Also

[pointsInPoly](#)

Examples

```
library(HRW)
x <- c(4,1,9,8,3,9,7)
y <- c(5,7,5,4,2,1,1)
## Not run: myBoundary <- createBoundary(x,y)
```

femSBMD

*Female spinal bone mineral densities***Description**

The femSBMD data frame has longitudinal data on spinal bone mineral density, age and ethnicity for female youths from a study on bone mineral acquisition.

Usage

```
data(femSBMD)
```

Format

This data frame contains the following columns:

idnum identification number unique to each subject.

spnbmd spinal bone mineral density in grams per square centimeter.

age age of subject in years.

ethnicity factor corresponding to the subject's ethnicity with levels Asian, Black, Hispanic and White.

black indicator of the subject being black:

0 = subject is black

1 = subject is not black.

hispanic indicator of the subject being Hispanic:

0 = subject is Hispanic

1 = subject is not Hispanic.

white indicator of the subject being white:

0 = subject is white

1 = subject is not white.

Source

Bachrach, L.K., Hastie, T., Wang, M.-C., Narasimhan, B. and Marcus, R. (1999). Bone mineral acquisition in healthy Asian, Hispanic, Black, and Caucasian youth: a longitudinal study. *Journal of Clinical Endocrinology and Metabolism*, **84**, 4702-4712.

Examples

```
library(HRW) ; data(femSBMD)
if (require("lattice"))
  print(xyplot(spnbmd ~ age|factor(ethnicity),groups = idnum,
              data = femSBMD,type = "b"))
```

growthIndianaAdolescent somatic growth in Indiana, U.S.A.

Description

Data on adolescent somatic growth obtained from a study of the mechanisms of human hypertension development conducted at the Indiana University School of Medicine, Indianapolis, Indiana, U.S.A. The data are restricted to a subset of 216 adolescents in the original study who had at least 9 height measurements. There are a total of 4,123 height measurements taken approximately every 6 months.

Usage

```
data(growthIndiana)
```

Format

A data frame with 4,123 observations on the following 5 variables:

`idnum` identification numbers of the 216 adolescents.

`height` height in centimeters.

`age` age in years.

`male` indicator of the adolescent being male:

1 = adolescent is male,

0 = adolescent is female.

`black` indicator of the adolescent being black:

1 = adolescent is black,

0 = adolescent is not black.

References

Pratt, J.H., Jones, J.J., Miller, J.Z., Wagner, M.A. and Fineberg, N.S. (1989). Racial differences in aldosterone excretion and plasma aldosterone concentrations in children. *New England Journal of Medicine*, **321**, 1152-1157.

Examples

```
library(HRW) ; data(growthIndiana)
growthINblackMales <- growthIndiana[(growthIndiana$male == 1) & (growthIndiana$black == 1),]
if (require("lattice"))
  xyplot(height ~ age|factor(idnum), data = growthINblackMales)
```

Description

Indonesian Children's Health Study of respiratory infections for a cohort of 275 Indonesian children. The data are longitudinal with each child having between 1 and 6 repeated measurements.

Usage

```
data(indonRespir)
```

Format

A data frame with 1200 observations on the following 12 variables:

`idnum` child identification number.

`respirInfec` indicator of presence of respiratory infection.

`age` age of the child in years.

`vitAdefic` indicator of Vitamin A deficiency:

1 = the child had Vitamin A deficiency,

0 = the child did not have Vitamin A deficiency.

`female` indicator of child being female:

1 = the child is female,

0 = the child is male.

`height` height of the child in centimeters.

`stunted` indicator of the child being "short for his/her age":

1 = the child is "short for his/her age",

0 = the child is not "short for his/her age"

`visit2` indicator that the child had exactly 2 clinical visits:

1 = the exact number of clinical visits was 2,

0 = the exact number of clinical visits was not 2.

`visit3` indicator that the child had exactly 3 clinical visits:

1 = the exact number of clinical visits was 3,

0 = the exact number of clinical visits was not 3.

`visit4` indicator that the child had exactly 4 clinical visits:

1 = the exact number of clinical visits was 4,

0 = the exact number of clinical visits was not 4.

`visit5` indicator that the child had exactly 5 clinical visits:

1 = the exact number of clinical visits was 5,

0 = the exact number of clinical visits was not 5.

`visit6` indicator that the child had exactly 6 clinical visits:

1 = the exact number of clinical visits was 6,

0 = the exact number of clinical visits was not 6.

Source

Sommer, A. (1982). *Nutritional Blindness*. New York: Oxford University Press.

References

Diggle, P., Heagerty, P., Liang, K.-L. and Zeger, S. (2002). *Analysis of Longitudinal Data (Second Edition)*. Oxford: Oxford University Press.

Examples

```
library(HRW) ; data(indonRespir)
if (require("mgcv"))
{
  fit <- gamm(respirInfec ~ s(age) + vitAdefic + female + height
              + stunted + visit2 + visit3 + visit4 + visit5 + visit6,
              random = list(idnum = ~1),family = binomial,data = indonRespir)
  summary(fit$gam)
}
```

lidar	<i>Light detection and ranging</i>
-------	------------------------------------

Description

The lidar data frame has 221 pairs from a Light Detection And Ranging (LIDAR) experiment.

Usage

```
data(lidar)
```

Format

This data frame contains the following columns:

`range` distance traveled before the light is reflected back to its source.

`logratio` logarithm of the ratio of received light from two laser sources.

Source

Sigrist, M. (Ed.) (1994). *Air Monitoring by Spectroscopic Techniques* (Chemical Analysis Series, vol. 197). New York: Wiley.

Examples

```
library(HRW) ; data(lidar)
plot(lidar$range,lidar$logratio)
```

ozoneSub*Ozone levels in midwest U.S.A.*

Description

This dataset is a subset of the ozone2 dataset in the fields package. It contains the 8-hour average ozone concentration at 147 sites in the midwest region of the U.S.

Usage

```
data(ozoneSub)
```

Format

A data frame with 147 observations on the following 3 variables:

longitude observation longitude.
latitude observation latitude.
ozone ozone level.

Source

Aerometric Information Retrieval System, the U.S. Environmental Protection Agency air quality data base.

References

Nychka, D., Furrer, R., Paige, J. and Sain, S. (2017). fields: Tools for spatial data. R package version 9.0. <https://www.r-project.org>.

Examples

```
library(HRW) ; data(ozoneSub)
if (require("mgcv"))
{
  fit.ozone.mgcv.tp <- gam(ozone ~ s(longitude,latitude,bs = "tp"),
                             data = ozoneSub,method = "REML")
  plot(fit.ozone.mgcv.tp,scheme = 2,
       main = "ozone concentration",bty = "l")
  points(ozoneSub$longitude,ozoneSub$latitude)
}
if (require("fields"))
  US(add = TRUE,lwd = 2)
```

plankton*Flow cytometric measurements on plankton organisms*

Description

The plankton data frame has data on six flow cytometric measurements for 400 plankton organisms categorized into 5 different species. The data are synthetic and were generated to test various machine learning algorithms for plankton species classification. More details are given in the 2001 ‘Cytometry’ article by Boddy, Wilkins and Morris (full reference below).

Usage

```
data(plankton)
```

Format

This data frame contains the following columns:

```
timeFlight time of flight.  
forwScatt forward-scatter.  
sideScatt side-scatter.  
redFluorBlueLight red fluorescence under blue light.  
greenFluorBlueLight green fluorescence under blue light.  
redFluorRedLight red fluorescence under red light.  
species name of the plankton species, which is either Dunaliella, Hemiselmis, Isochrysis, Pavlova or Pyramimonas.
```

Source

Boddy, L., Wilkins, M.F. and Morris, C.W. (2001). Pattern recognition in flow cytometry. *Cytometry*, **44**, 195-209.

Examples

```
library(HRW) ; data(plankton)  
pointCols <- c("red", "blue", "green", "orange", "purple")  
pairs(plankton[, 1:6], col = pointCols[plankton$species], pch = ".")
```

pointsInPoly*Points inside/outside polygon determination*

Description

Determination of whether each member of a set of bivariate points are inside a polygon.

Usage

```
pointsInPoly(pointsCoords, polygon)
```

Arguments

pointsCoords	two-column matrix with each row specifying the (x,y) coordinates of a point.
polygon	two-column matrix with each row specifying the (x,y) coordinates of a vertex of a polygon.

Details

Geometric results are used to determine whether each of a set of bivariate points is inside or outside a polygon. A Boolean vector of indicators of whether or not each point is inside the polygon is returned.

Value

A Boolean array with length equal to the number of rows in 'pointsCoords' corresponding to whether or not each of the corresponding points in 'pointCoords' are inside 'polygon'.

Author(s)

M.P. Wand <matt.wand@uts.edu.au>

See Also

[createBoundary](#)

Examples

```
library(HRW)
myPolygon <- rbind(c(1,9),c(8,8),c(9,3),c(3,2),c(1,9))/10
plot(0:1,0:1,type = "n") ; lines(myPolygon)
xyMat <- cbind(runif(10),runif(10))
inPoly <- pointsInPoly(xyMat,myPolygon) ; print(inPoly)
points(xyMat[,1],xyMat[,2],col = as.numeric(inPoly) + 2)
```

protein	<i>Protein intake dietary study</i>
---------	-------------------------------------

Description

The protein data frame has longitudinal data on protein intake, body mass index and age of subjects in a dietary study.

Usage

```
data(protein)
```

Format

This data frame contains the following columns:

`idnum` identification number unique to each subject.

`proteinBioM` logarithm of intake of protein as measured by the biomarker urinary.

`age` age of subject in years.

`BMI` body mass index.

`proteinRecall` logarithm of intake of protein as measured by a 24-hour recall instrument.

`female` indicator that subject is female:

1=subject is female,

0=subject is male.

Source

Kipnis, V., Subar, A.F., Midthune, D., Freedman, L.S., Ballard-Barbash, R., Troiano, R., Bingham, S., Schoeller, D.A., Schatzkin, A. and Carroll, R.J. (2003). The structure of dietary measurement error: results of the OPEN biomarker study. *American Journal of Epidemiology*, **158**, 14-21.

Examples

```
library(HRW) ; data(protein)
if (require("lattice"))
  print(xyplot(proteinBioM ~ BMI | factor(female), groups = idnum,
              data = protein, type = "b"))
```

ragweed

Ragweed pollen in Kalamazoo, U.S.A.

Description

The `ragweed` data frame has data on ragweed levels and meteorological variables for 334 days in Kalamazoo, Michigan, U.S.A.

Usage

```
data(ragweed)
```

Format

This data frame contains the following columns:

`pollenCount` ragweed pollen count (grains per cubic metre).
`year` one of 1991, 1992, 1993 or 1994.
`dayInSeason` day number in the current ragweed pollen season.
`temperature` temperature for the corresponding day (degrees Fahrenheit).
`temperatureResidual` residual from fitting a 5 effective degrees of freedom smoothing splines to temperature versus day number for each annual ragweed pollen season (degrees Fahrenheit).
`rain` indicator of significant rain on the corresponding day:
 1=at least 3 hours of steady or brief but intense rain,
 0=otherwise.
`windSpeed` wind speed for the corresponding day (knots).

Source

Stark, P. C., Ryan, L. M., McDonald, J. L. and Burge, H. A. (1997). Using meteorologic data to model and predict daily ragweed pollen levels. *Aerobiologia*, **13**, 177-184.

References

Ruppert, D., Wand, M.P. and Carroll, R.J. (2003). *Semiparametric Regression* Cambridge University Press.

Examples

```
library(HRW) ; data(ragweed)
pairs(ragweed,pch = ".")
```

scallop*Scallop abundance off Long Island, U.S.A.*

Description

The scallop data frame has 148 triplets concerning scallop abundance; based on a 1990 survey cruise in the Atlantic continental shelf off Long Island, New York, U.S.A.

Usage

```
data(scallop)
```

Format

This data frame contains the following columns:

`latitude` degrees latitude (north of the Equator).

`longitude` degrees longitude (west of Greenwich).

`totalCatch` total size of scallop catch at location specified by 'latitude' and 'longitude'.

Source

Ecker, M.D. and Heltshe, J.F. (1994). Geostatistical estimates of scallop abundance. In *Case Studies in Biometry*. Lange, N., Ryan, L., Billard, L., Brillinger, D., Conquest, L. and Greenhouse, J. (eds.) New York: John Wiley & Sons, 107-124.

References

Ruppert, D., Wand, M.P. and Carroll, R.J. (2003). *Semiparametric Regression*. Cambridge University Press.

Examples

```
library(HRW) ; data(scallop)
pairs(scallop)
```

schoolResults	<i>School results in the United Kingdom</i>
---------------	---

Description

The `schoolResults` data frame has multilevel data school results and gender for 1,905 school children from 73 schools in United Kingdom.

Usage

```
data(schoolResults)
```

Format

This data frame contains the following columns:

`schoolID` school identification number.

`studentID` student identification number.

`female` indicator that child is female:

1=child is female,
0=child is male.

`writtenScore` score on traditional written examination paper out of a total of 160.

`courseScore` score from projects undertaken during the course and marked by the student's own teacher, out of a total of 108.

Source

Creswell, M. (1991). A multilevel bivariate model. In *Data Analysis with ML3* (eds. Prosser, R., Rasbash, J. and Goldstein, H.) London: Institute of Education, London.

Examples

```
library(HRW) ; data(schoolResults)
if (require("lattice"))
  print(xyplot(writtenScore ~ courseScore | factor(schoolID),
    groups = female, data = schoolResults))
```

summMCMC	<i>Summarizes Markov chain Monte Carlo (MCMC) samples both graphically and numerically</i>
----------	--

Description

Given a set of MCMC for possibly several parameters the following summaries are produced: 1. a trace (time series) plot of each MCMC, 2. a plot of each MCMC sample against the 1-lagged sample, 3. estimated autocorrelation function (acf), 4. Brooks-Gelman-Rubin (BGR) diagnostic plot in cases where multiple chains are inputted, 5. kernel density estimate of the posterior density function based on the MCMC sample, 6. numerical summary consisting of the MCMC-based estimates of posterior means and credible sets for each parameter.

Usage

```
summMCMC(xList, EPSfileName, PDFfileName, plotInd = 1, parNames, columnHeadings,
           colourVersion = TRUE, credLevel = 0.95, columnHeadCex = NULL, paletteNum = 1,
           numerSummCex = NULL, BGRsttPos = 10, BGRyRange = c(0.95, 1.25), BGRtickPos = 1.2,
           BGRlogTransf, BGRlogitTransf, KDExlim, KDEvertLine = TRUE, KDEvertLineCol = "black",
           addTruthToKDE = NULL)
```

Arguments

xList	list of matrices, where each matrix corresponds to a different chain, and the columns of each matrix correspond to different parameters. The matrices each have dimension "numMCMC" by "numParms"; where "numMCMC" is the size of the MCMC sample and "numParms" is the number of parameters being summarized.
EPSfileName	filename if the summary is to be saved as a (encapsulated) Postscript file. If this argument and 'PDFfileName' are both not specified then the summary is printed to the screen.
PDFfileName	filename if the summary is to be saved as a PDF file. If this argument and 'EPSfileName' are both not specified then the summary is printed to the screen.
plotInd	if "numChains" exceeds 1 then this indicates which chain is summarized in the non-BGR panels. The BGR panels are Brooks-Gelman-Rubin diagnostic plots are use all chains. The default value is 1.
parNames	list containing a vector of character strings for the parameter names. The maximum length of the vector is 3.
columnHeadings	vector containing column headings. The default is: c("parameter", "trace", "lag 1", "acf", "BGR", "density", "summary").
colourVersion	logical flag indicating if summary should be in colour. The default is TRUE.
credLevel	number between 0 and 1 specifying the credible set level. The default is 0.95.
columnHeadCex	positive number specifying character expansion factor for the column headings.
paletteNum	either 1 or 2 to specify the colour palette to be used.

numerSummCex	positive number specifying character expansion factor for the numerical summary (last column).
BGRsttPos	starting position for the Brooks-Gelman-Rubin plots. The default value is 10.
BGRyRange	vertical axis limits for the Brooks-Gelman-Rubin plots. The default value is c(0.95,1.25).
BGRtickPos	position of tick mark on vertical axis for the Brooks-Gelman-Rubin plots. The default value is 1.2.
BGRlogTransf	vector containing indices of those parameters for which the Brooks-Gelman-Rubin plots should be done on a logarithmic scale.
BGRlogitTransf	vector containing indices of those parameters for which the Brooks-Gelman-Rubin plots should be done on a logit scale.
KDExlim	list of vectors of length 2 specifying the horizontal axis limits for the kernel density estimates.
KDEvertLine	logical flag indicating if a vertical line at zero should be added to the kernel density estimates. The default value is TRUE.
KDEvertLineCol	Colour of the vertical line at zero for kernel density estimates. The default value is "black".
addTruthToKDE	Vector indicating 'true' values of parameters. The default value is NULL. If 'addTruthToKDE' is non-NULL then dashed vertical lines corresponding to true values (if known from simulation) are added.

Author(s)

Matt Wand <matt.wand@uts.edu.au>

Examples

```
library(HRW)
xListSingleChain <- list(cbind(rnorm(100),rnorm(100),rnorm(100),rnorm(100)))
summMCMC(xListSingleChain,parNames = list("par1","par2","par3","par4"))
xListMultipleChains <- list(chain1 = cbind(rnorm(100),rnorm(100),rnorm(100),rnorm(100)),
                                chain2 = cbind(rnorm(100),rnorm(100),rnorm(100),rnorm(100)))
summMCMC(xListMultipleChains,parNames = list("par1","par2","par3","par4"))
```

Description

The SydneyRealEstate data frame has data on several variables concerning houses sold in Sydney, Australia, during 2001.

Usage

```
data(SydneyRealEstate)
```

Format

This data frame contains the following columns:

`logSalePrice` natural logarithm of sale price in Australian dollars.
`lotSize` lot size in square meters but with some imputation.
`longitude` degrees longitude of location of house.
`latitude` degrees latitude of location of house.
`saleDate` sale date in dd/mm/yy format.
`saleQtr` financial quarter in which sale took place.
`infRate` inflation rate measure as a percentage.
`postCode` four-digit post code of the suburb in which the house located.
`crimeDensity` crime density measure for the suburb in which the house is located.
`crimeRate` crime rate measure for the suburb in which the house is located.
`income` average weekly income of the suburb in which the house is located.
`distToBusStop` distance from house to the nearest bus stop (kilometers).
`distToCoastline` distance from house to the nearest coastline location (kilometers).
`distToNatPark` distance from house to the nearest national park (kilometers).
`distToPark` distance from house to the nearest park (kilometers).
`distToRailLine` distance from house to the nearest railway line (kilometers).
`distToRailStation` distance from house to the nearest railway station (kilometers).
`distToHighway` distance from house to the nearest highway (kilometers).
`distToFreeway` distance from house to the nearest freeway (kilometers).
`distToTunnel` distance from house to the Sydney Harbour Tunnel (kilometers).
`distToMainRoad` distance from house to the nearest main road (kilometers).
`distToSealedRoad` distance from house to the nearest sealed road (kilometers).
`distToUnsealedRoad` distance from house to the nearest sealed road (kilometers).
`airNoise` aircraft noise exposure measure.
`foreignerRatio` proportion of foreigners in the suburb in which the house is located.
`distToGPO` distance from the house to the General Post Office in Sydney's central business district (kilometers).
`NO` nitrous oxide level recorded at the air pollution monitoring station nearest to the house.
`NO2` nitrogen dioxide level recorded at the air pollution monitoring station nearest to the house.
`ozone` ozone level recorded at the air pollution monitoring station nearest to the house.
`neph` nephelometer suspended matter measurement recorded at the air pollution monitoring station nearest to the house.
`PM10` particulate matter with a diameter of under 10 micrometers level recorded at the air pollution monitoring station nearest to the house.
`SO2` sulphur dioxide level recorded at the air pollution monitoring station nearest to the house.

distToAmbulance distance from house to the nearest ambulance station (kilometers).
 distToFactory distance from house to the nearest factory (kilometers).
 distToFerry distance from house to the nearest ferry wharf (kilometers).
 distToHospital distance from house to the nearest hospital (kilometers).
 distToMedical distance from house to the nearest medical services (kilometers).
 distToSchool distance from house to the nearest school (kilometers).
 distToUniversity distance from house to the nearest university (kilometers).

References

Chernih, A. and Sherris, M. (2004). Geoadditive hedonic pricing models. Unpublished manuscript. University of New South Wales, Australia.

Examples

```

library(HRW) ; data(SydneyRealEstate)
## Not run:
for (colNum in setdiff((2:39),c(5,8)))
{
  plot(jitter(SydneyRealEstate[,colNum]),SydneyRealEstate$logSalePrice,pch = ".",
        xlab = names(SydneyRealEstate)[colNum],ylab = "log(sale price)",col = "blue")
  readline("Hit Enter to continue.\n")
}
## End(Not run)

```

SydneyRealEstateBdry *Polygonal boundary of Sydney*

Description

A two-column data frame containing the (longitude,latitude) pairs for the vertices of a 202-sided polygon. The polygon was created manually using the HRW package function `createBdry()`. The polygon tightly encompasses the majority of the (longitude,latitude) data of the HRW package ‘SydneyRealEstate’ data frame and approximately corresponds to the residential parts of Sydney, Australia.

Usage

```
data(SydneyRealEstateBdry)
```

Format

A data frame with 203 observations on the following 2 variables (note that the first vertex is repeated at the end of the data frame):

longitude longitudinal position of a vertex.
 latitude latitudinal position of the same vertex.

See Also

[SydneyRealEstate](#)

Examples

```
library(HRW) ; data(SydneyRealEstateBdry) ; data(SydneyRealEstate)
plot(SydneyRealEstate$longitude,SydneyRealEstate$latitude,cex = 0.1)
lines(SydneyRealEstateBdry,lwd = 5,col = "red")
```

TreasuryRate

U.S. Treasury rate

Description

One-month maturity U.S. Treasury rate during the period 2001-2013.

Usage

```
data(TreasuryRate)
```

Format

A data frame with 3,117 observations on each of the following 2 variables:

date days from July 31, 2001 until July 10, 2013 with the occasional missing values due to holidays.

rate daily one-month maturity U.S. Treasury rate.

Source

Federal Reserve Bank of St. Louis, U.S.A.

Examples

```
library(HRW) ; data(TreasuryRate)
TRdate <- as.Date(TreasuryRate$date,"%m/%d/%Y")[!is.na(TreasuryRate$rate)]
TRrate <- TreasuryRate$rate[!is.na(TreasuryRate$rate)]
plot(TRdate,TRrate,type = "l",bty = "1",xlab = "date",ylab = "U.S. Treasury rate")
```

UtahPEF

*Peak expiratory flow in Utah, U.S.A.***Description**

The UtahPEF data frame data contains longitudinal data on peak expiratory flow, air pollution and temperature for a cohort of 41 schoolchildren in the Utah Valley, U.S.A., during 1990-1991.

Usage

```
data(UtahPEF)
```

Format

This data frame contains the following columns:

`idnum` schoolchild identification number.

`devPEF` daily peak expiratory flow measurements for each schoolchild minus the overall average for that schoolchild.

`PM10withMA5` 5-day moving average of the concentration of particulate matter 10 micrometers or less in diameter.

`lowTemp` lowest temperature in degrees Fahrenheit on the day of recording.

`timeTrend` day of the study on which the measurements were made.

Source

Pope, C.A., Dockery, D.W., Spengler, J.D. and Raizenne, M.E. (1991). Respiratory health and PM₁₀ pollution: a daily time series analysis. *American Review of Respiratory Disease*, **144**, 668-674.

Examples

```
library(HRW) ; data(UtahPEF)
if (require("lattice"))
  print(xyplot(devPEF ~ PM10withMA5|as.factor(idnum), data = UtahPEF))
```

Description

'WarsawApts' is a subset of the data set 'apartments' in the R package PBImisc. This dataset contains the prices of the apartments which were sold in Warsaw, Poland, during the calendar years 2007 to 2009.

Usage

```
data(WarsawApts)
```

Format

A data frame with 409 observations on the following 6 variables:

surface area of the apartment in square meters.
 district a factor corresponding to the district of Warsaw with levels Mokotow, Srodmiescie, Wola and Zoliborz.
 n.rooms number of rooms in the apartment.
 floor floor on which the apartment is located.
 construction.date year that the apartment was constructed.
 areaPerMzloty area in square meters per million zloty.

Source

The Polish real estate web-site <https://www.oferty.net>.

References

Biecek, P. (2016). PBImisc: A set of datasets in My Classes or in the Book 'Modele Liniowe i Mieszane w R, Wraz z Przykladami w Analizie Danych' 1.0.

Examples

```
library(HRW) ; data(WarsawApts)
x <- WarsawApts$construction.date
y <- WarsawApts$areaPerMzloty
plot(x,y,bty = "l",col = "dodgerblue")
if (require("mgcv"))
{
  fitGAMcr <- gam(y ~ s(x,bs = "cr",k = 30))
  xg <- seq(min(x),max(x),length = 1001)
  fHatgGAMcr <- predict(fitGAMcr,newdata = data.frame(x = xg))
  lines(xg,fHatgGAMcr,col = "darkgreen")
}
```

yields

U.S., European and Japanese yield curves

Description

U.S., European and Japanese yield curves. These are functions of maturity. This dataset has 91 columns. The first column is the date, columns 2 to 31 are European yields at maturities from 1 to 30 years, columns 32 to 61 are Japanese yields at these maturities, and columns 62 to 91 are U.S. yields at the same maturities.

Usage

```
data(yields)
```

Format

A data frame with 1565 observations on the following 91 variables:

date date when the yield is measured.
EU01 European yield at a maturity of 1 year.
EU02 European yield at a maturity of 2 years.
EU03 European yield at a maturity of 3 years.
EU04 European yield at a maturity of 4 years.
EU05 European yield at a maturity of 5 years.
EU06 European yield at a maturity of 6 years.
EU07 European yield at a maturity of 7 years.
EU08 European yield at a maturity of 8 years.
EU09 European yield at a maturity of 9 years.
EU10 European yield at a maturity of 10 years.
EU11 European yield at a maturity of 11 years.
EU12 European yield at a maturity of 12 years.
EU13 European yield at a maturity of 13 years.
EU14 European yield at a maturity of 14 years.
EU15 European yield at a maturity of 15 years.
EU16 European yield at a maturity of 16 years.
EU17 European yield at a maturity of 17 years.
EU18 European yield at a maturity of 18 years.
EU19 European yield at a maturity of 19 years.
EU20 European yield at a maturity of 20 years.
EU21 European yield at a maturity of 21 years.

- EU22 European yield at a maturity of 22 years.
- EU23 European yield at a maturity of 23 years.
- EU24 European yield at a maturity of 24 years.
- EU25 European yield at a maturity of 25 years.
- EU26 European yield at a maturity of 26 years.
- EU27 European yield at a maturity of 27 years.
- EU28 European yield at a maturity of 28 years.
- EU29 European yield at a maturity of 29 years.
- EU30 European yield at a maturity of 30 years.
- JP01 Japanese yield at a maturity of 1 year.
- JP02 Japanese yield at a maturity of 2 years.
- JP03 Japanese yield at a maturity of 3 years.
- JP04 Japanese yield at a maturity of 4 years.
- JP05 Japanese yield at a maturity of 5 years.
- JP06 Japanese yield at a maturity of 6 years.
- JP07 Japanese yield at a maturity of 7 years.
- JP08 Japanese yield at a maturity of 8 years.
- JP09 Japanese yield at a maturity of 9 years.
- JP10 Japanese yield at a maturity of 10 years.
- JP11 Japanese yield at a maturity of 11 years.
- JP12 Japanese yield at a maturity of 12 years.
- JP13 Japanese yield at a maturity of 13 years.
- JP14 Japanese yield at a maturity of 14 years.
- JP15 Japanese yield at a maturity of 15 years.
- JP16 Japanese yield at a maturity of 16 years.
- JP17 Japanese yield at a maturity of 17 years.
- JP18 Japanese yield at a maturity of 18 years.
- JP19 Japanese yield at a maturity of 19 years.
- JP20 Japanese yield at a maturity of 20 years.
- JP21 Japanese yield at a maturity of 21 years.
- JP22 Japanese yield at a maturity of 22 years.
- JP23 Japanese yield at a maturity of 23 years.
- JP24 Japanese yield at a maturity of 24 years.
- JP25 Japanese yield at a maturity of 25 years.
- JP26 Japanese yield at a maturity of 26 years.
- JP27 Japanese yield at a maturity of 27 years.
- JP28 Japanese yield at a maturity of 28 years.

- JP29 Japanese yield at a maturity of 29 years.
- JP30 Japanese yield at a maturity of 30 years.
- US01 U.S. yield at a maturity of 1 year.
- US02 U.S. yield at a maturity of 2 years.
- US03 U.S. yield at a maturity of 3 years.
- US04 U.S. yield at a maturity of 4 years.
- US05 U.S. yield at a maturity of 5 years.
- US06 U.S. yield at a maturity of 6 years.
- US07 U.S. yield at a maturity of 7 years.
- US08 U.S. yield at a maturity of 8 years.
- US09 U.S. yield at a maturity of 9 years.
- US10 U.S. yield at a maturity of 10 years.
- US11 U.S. yield at a maturity of 11 years.
- US12 U.S. yield at a maturity of 12 years.
- US13 U.S. yield at a maturity of 13 years.
- US14 U.S. yield at a maturity of 14 years.
- US15 U.S. yield at a maturity of 15 years.
- US16 U.S. yield at a maturity of 16 years.
- US17 U.S. yield at a maturity of 17 years.
- US18 U.S. yield at a maturity of 18 years.
- US19 U.S. yield at a maturity of 19 years.
- US20 U.S. yield at a maturity of 20 years.
- US21 U.S. yield at a maturity of 21 years.
- US22 U.S. yield at a maturity of 22 years.
- US23 U.S. yield at a maturity of 23 years.
- US24 U.S. yield at a maturity of 24 years.
- US25 U.S. yield at a maturity of 25 years.
- US26 U.S. yield at a maturity of 26 years.
- US27 U.S. yield at a maturity of 27 years.
- US28 U.S. yield at a maturity of 28 years.
- US29 U.S. yield at a maturity of 29 years.
- US30 U.S. yield at a maturity of 30 years.

Source

European Central Bank (European yields), Japanese Ministry of Finance (Japanese yields) and U.S. Federal Reserve Board (U.S. yields).

Examples

```
library(HRW) ; data(yields)
t <- 1:30 ; yieldsCleaned <- na.omit(yields)[,-1]
plot(t,yieldsCleaned[1,61:90], type="l",ylim = c(0,6),lwd = 2,
      xlab = "maturity (years)",ylab = "yield",
      bty = "l",cex.lab = 1.5,cex.axis = 1.5)
for (i in 2:14) lines(t,yieldsCleaned[100*i+1,61:90],col = i,lwd = 2)
```

ZOSull

O'Sullivan spline design matrices

Description

Constructs a design matrix consisting of O'Sullivan cubic spline functions of an array of abscissae. Typically the array corresponds to either observed values of a predictor or an abscissa grid for plotting purposes.

Usage

```
ZOSull(x,range.x,intKnots,drv = 0)
```

Arguments

<code>x</code>	array of abscissae.
<code>range.x</code>	array of length 2 such that <code>range.x[1] <= min(x)</code> and <code>range.x[2] >= max(x)</code> .
<code>intKnots</code>	ordered array of length smaller than that of <code>x</code> and consisting of unique numbers between <code>min(x)</code> and <code>max(x)</code> that specifies the positions of internal knots, that define the spline basis (see the Wand and Ormerod (2008) reference below for full mathematical details).
<code>drv</code>	either 0,1 or 2 with a default value of 0. If <code>drv = 1</code> then the first derivatives of the O'Sullivan spline basis functions are computed instead. Similarly, if <code>drv = 2</code> then the second derivatives are computed.

Value

A matrix of with `length(x)` rows and `(length(intKnots) + 2)` columns with each column containing a separate O'Sullivan spline basis function (determined by `range.x` and `intKnots`) of the abscissae in `x`. The values of `range.x` and `intKnots` are included as attributes of the fit object.

Author(s)

Matt Wand <matt.wand@uts.edu.au>

References

O'Sullivan, F. (1986). A statistical perspective on ill-posed inverse problems (with discussion). *Statistical Science*, **1**, 505-527.

Wand, M.P. and Ormerod, J.T. (2008). On semiparametric regression with O'Sullivan penalized splines. *Australian and New Zealand Journal of Statistics*. **50**, 179-198.

Examples

```
library(HRW)
x <- WarsawApts$construction.date
a <- 1.01*min(x) - 0.01*max(x)
b <- 1.01*max(x) - 0.01*min(x) ; numIntKnots <- 23
intKnots <- quantile(unique(x),seq(0,1,length = (numIntKnots + 2))[-c(1,(numIntKnots + 2))])
xg <- seq(a,b,length = 1001)
Zg <- ZOSull(xg,range.x = c(a,b),intKnots = intKnots)
plot(0,type = "n",xlim = range(xg),ylim = range(Zg),
     bty = "l",xlab = "construction date (year)",
     ylab = "spline basis function")
for (k in 1:ncol(Zg)) lines(xg,Zg[,k],col = k,lwd = 2)
  lines(c(min(xg),max(xg)),rep(0,2),col = "darkmagenta")
for (k in 1:numIntKnots)
  points(intKnots[k],0,pch = 18,cex = 2,col = "darkmagenta")
```

Index

BanglaContrac, 2
BCR, 3
BostonMortgages, 4
brainImage, 5
capm, 6
carAuction, 7
CHD, 9
coral, 10
createBoundary, 11, 18
femSBMD, 12
growthIndiana, 13
indonRespir, 14
lidar, 15
ozoneSub, 16
plankton, 17
pointsInPoly, 11, 18
protein, 19
ragweed, 20
scallop, 21
schoolResults, 22
summMCMC, 23
SydneyRealEstate, 24
SydneyRealEstateBdry, 26
TreasuryRate, 27
UtahPEF, 28
WarsawApts, 29
yields, 30
Z0Sull, 33