Package ‘DAAGxtras’

February 19, 2015

Version 0.8-4

Date 2013-October-16

Title Data Sets and Functions, supplementary to DAAG

Author John Maindonald

Maintainer John Maindonald <john.maindonald@anu.edu.au>

Description various data sets used in additional exercises for
the book Maindonald, J.H. and Braun, W.J. (3rd edn 2010)
``Data Analysis and Graphics Using R'', and for a
'Data Mining' course. Note that a number of datasets
that were in earlier versions of this package have been
transferred to the DAAG package.

LazyLoad true

LazyData true

Depends R (>= 2.10.0)

Suggests DAAG, MASS, lattice, randomForest

ZipData yes

License Unlimited

URL http://www.maths.anu.edu.au/~johnm

NeedsCompilation no

Repository CRAN

Date/Publication 2013-10-16 10:46:09

R topics documented:

`DAAGxtras-package` *The R DAAGxtras Package*

Description

various data sets used in additional exercises for the book Maindonald, J.H. and Braun, W.J. (3rd edn 2010) "Data Analysis and Graphics Using R", and/or used for various instructional purposes.

Details

Note especially the datasets `rockArt` (multivariate binary data on Pacific rock art), `audists` (Australian road distances), `greatLakes` (Great Lakes lake levels, by year). and the dataset `fishRivers` that has information on preferred fish river types.

For a complete list, use `library(help="DAAGxtras")`.

Author(s)

Author: John Maindonald

Maintainer: John Maindonald `<john.maindonald@anu.edu.au>`

`compareModels` *Compare accuracy of alternative classification methods*

Description

Compare, between models, probabilities that the models assign to membership in the correct group or class. Probabilites should be estimated from cross-validation or from bootstrap out-of-bag data or preferably for test data that are completely separate from the data used to dervive the model.

Usage

```
compareModels(groups = fgl$type, estprobs = list(lda = NULL, rf = NULL),
             gpnames = NULL, robust = TRUE, print = TRUE)
```

Arguments

`groups`	Factor that specifies the groups
`estprobs`	List whose elements (with names that identify the models) are matrices that give for each observation (row) estimated probabilities of membership for each of the groups (columns).
`gpnames`	Character: names for groups, if different from `levels(groups)`
`robust`	Logical, `TRUE` or `FALSE`
`print`	Logical. Should results be printed?

Details

The estimated probabilities are compared directly, under normal distribution assumptions. An effect is fitted for each observation, plus an effect for the method. Comparison on a logit scale may sometimes be preferable. An option to allow this is scheduled for incorporation in a later version.

Value

`modelAVS`	Average accuracies for models
`modelSE`	Approximate average SE for comparing models
`gpAVS`	Average accuracies for groups
`gpSE`	Approximate average SE for comparing groups
`obsEff`	Effects assigned to individual observations

Note

The analysis estimates effects due to model and group (gp), after accounting for differences between observations.

Author(s)

John Maindonald

Examples

```
library(MASS)
library(DAAG)
ldahat <- lda(species ~ length+breadth, data=cuckoos, CV=TRUE)$posterior
qdahat <- qda(species ~ length+breadth, data=cuckoos, CV=TRUE)$posterior
compareModels(groups=cuckoos$species, estprobs=list(lda=ldahat,
              qda=qdahat), robust=FALSE)
rfOUT <- try(require(randomForest, quietly=TRUE))
rfOUT.log <- is.logical(rfOUT)
if ((rfOUT.log==TRUE)&(rfOUT==TRUE)){
rfhat <- predict(randomForest(species ~ length+breadth, data=cuckoos),
                 type="prob")
compareModels(groups=cuckoos$species, estprobs=list(lda=ldahat,
              qda=qdahat, rf=rfhat), robust=FALSE)
}
```

`covsample` *Sample of UCI Machine Learning Forest Cover Dataset*

Description

Forest cover type is recorded, for every 50th observation taken from 581012 observations in the original dataset, together with a physical geographical variables that may account for the forest cover type.

Usage

```
data(covsample)
```

Format

A data frame with 11318 observations on the following 55 variables.

`V1` a numeric vector

`V2` a numeric vector

`V3` a numeric vector

`V4` a numeric vector

`V5` a numeric vector

`V6` a numeric vector

`V7` a numeric vector

`V8` a numeric vector

`V9` a numeric vector

`V10` a numeric vector

`V11` a numeric vector

`V12` a numeric vector

`V13` a numeric vector

`V14` a numeric vector

`V15` a numeric vector

`V16` a numeric vector

`V17` a numeric vector

`V18` a numeric vector

`V19` a numeric vector

`V20` a numeric vector

`V21` a numeric vector

`V22` a numeric vector

`V23` a numeric vector

V24 a numeric vector

V25 a numeric vector

V26 a numeric vector

V27 a numeric vector

V28 a numeric vector

V29 a numeric vector

V30 a numeric vector

V31 a numeric vector

V32 a numeric vector

V33 a numeric vector

V34 a numeric vector

V35 a numeric vector

V36 a numeric vector

V37 a numeric vector

V38 a numeric vector

V39 a numeric vector

V40 a numeric vector

V41 a numeric vector

V42 a numeric vector

V43 a numeric vector

V44 a numeric vector

V45 a numeric vector

V46 a numeric vector

V47 a numeric vector

V48 a numeric vector

V49 a numeric vector

V50 a numeric vector

V51 a numeric vector

V52 a numeric vector

V53 a numeric vector

V54 a numeric vector

V55 a numeric vector

For details, see http://kdd.ics.uci.edu/databases/covertype/covertype.data.html

Details

For detailed information on the UCI dataset, see http://kdd.ics.uci.edu/databases/covertype/covertype.data.html

Variables V1 to V54 are physical geographical variables. Variable V55 is cover type, one of types 1 - 7.

Note the omission of any information on geographical location. Distance through the data seems however to be, in part, a proxy for geographical location.

Source

http://kdd.ics.uci.edu/databases/covertype/covertype.html

References

Blackard, Jock A. 1998. "Comparison of Neural Networks and Discriminant Analysis in Predicting Forest Cover Types." Ph.D. dissertation. Department of Forest Sciences. Colorado State University. Fort Collins, Colorado.

Examples

```
data(covsample)
options(digits=3)
tab.sample <- table(covsample$V55)
tab.sample/sum(tab.sample)
rm(covsample)
data(covtrain)
tab.train <- table(covtrain$V55)
tab.train/sum(tab.train)
rm(covtrain)
data(covtest)
tab.test <- table(covtest$V55)
tab.test/sum(tab.test)
rm(covtest)
```

---

covtest *Sample of UCI Machine Learning Forest Cover Dataset*

Description

Dataset used as test data in the study cited below. These are observations 11341 to 15120, out of 581012, in the dataset on the UCI site. Forest cover type is recorded, together with information on physical geographical variables that may account for the forest cover type.

Usage

```
data(covtest)
```

Format

A data frame with 11318 observations on the following 55 variables.

V1 a numeric vector

V2 a numeric vector

V3 a numeric vector

V4 a numeric vector

V5 a numeric vector

V6 a numeric vector

V7 a numeric vector

V8 a numeric vector

V9 a numeric vector

V10 a numeric vector

V11 a numeric vector

V12 a numeric vector

V13 a numeric vector

V14 a numeric vector

V15 a numeric vector

V16 a numeric vector

V17 a numeric vector

V18 a numeric vector

V19 a numeric vector

V20 a numeric vector

V21 a numeric vector

V22 a numeric vector

V23 a numeric vector

V24 a numeric vector

V25 a numeric vector

V26 a numeric vector

V27 a numeric vector

V28 a numeric vector

V29 a numeric vector

V30 a numeric vector

V31 a numeric vector

V32 a numeric vector

V33 a numeric vector

V34 a numeric vector

V35 a numeric vector

V36 a numeric vector

V37 a numeric vector

V38 a numeric vector

V39 a numeric vector

V40 a numeric vector

V41 a numeric vector

V42 a numeric vector

V43 a numeric vector

V44 a numeric vector

V45 a numeric vector

V46 a numeric vector

V47 a numeric vector

V48 a numeric vector

V49 a numeric vector

V50 a numeric vector

V51 a numeric vector

V52 a numeric vector

V53 a numeric vector

V54 a numeric vector

V55 a numeric vector

For details, see http://kdd.ics.uci.edu/databases/covertype/covertype.data.html

Details

For further details, see http://kdd.ics.uci.edu/databases/covertype/covertype.data.html. Note the omission of any information on geographical location. Distance through the data seems however to be, in part, a proxy for geographical location.

Source

http://kdd.ics.uci.edu/databases/covertype/covertype.html

References

Blackard, Jock A. 1998. "Comparison of Neural Networks and Discriminant Analysis in Predicting Forest Cover Types." Ph.D. dissertation. Department of Forest Sciences. Colorado State University. Fort Collins, Colorado.

Examples

```
data(covtest)
```

`covtrain` *Sample of UCI Machine Learning Forest Cover Dataset*

Description

Dataset used as training data in the study cited below. These are the first 11,340 observations, out of 581012, in the dataset on the UCI site. Forest cover type is recorded, together with information on physical geographical variables that may account for the forest cover type.

Usage

```
data(covtrain)
```

Format

A data frame with 11318 observations on the following 55 variables.

`V1` a numeric vector

`V2` a numeric vector

`V3` a numeric vector

`V4` a numeric vector

`V5` a numeric vector

`V6` a numeric vector

`V7` a numeric vector

`V8` a numeric vector

`V9` a numeric vector

`V10` a numeric vector

`V11` a numeric vector

`V12` a numeric vector

`V13` a numeric vector

`V14` a numeric vector

`V15` a numeric vector

`V16` a numeric vector

`V17` a numeric vector

`V18` a numeric vector

`V19` a numeric vector

`V20` a numeric vector

`V21` a numeric vector

`V22` a numeric vector

`V23` a numeric vector

V24 a numeric vector

V25 a numeric vector

V26 a numeric vector

V27 a numeric vector

V28 a numeric vector

V29 a numeric vector

V30 a numeric vector

V31 a numeric vector

V32 a numeric vector

V33 a numeric vector

V34 a numeric vector

V35 a numeric vector

V36 a numeric vector

V37 a numeric vector

V38 a numeric vector

V39 a numeric vector

V40 a numeric vector

V41 a numeric vector

V42 a numeric vector

V43 a numeric vector

V44 a numeric vector

V45 a numeric vector

V46 a numeric vector

V47 a numeric vector

V48 a numeric vector

V49 a numeric vector

V50 a numeric vector

V51 a numeric vector

V52 a numeric vector

V53 a numeric vector

V54 a numeric vector

V55 a numeric vector

For details, see http://kdd.ics.uci.edu/databases/covertype/covertype.data.html

Details

For details, see http://kdd.ics.uci.edu/databases/covertype/covertype.data.html. Note the omission of any information on geographical location. Distance through the data seems however to be, in part, a proxy for geographical location.

Source

http://kdd.ics.uci.edu/databases/covertype/covertype.html

References

Blackard, Jock A. 1998. "Comparison of Neural Networks and Discriminant Analysis in Predicting Forest Cover Types." Ph.D. dissertation. Department of Forest Sciences. Colorado State University. Fort Collins, Colorado.

Examples

```
data(covtrain)
```

---

earlycrcktr — *Lifespans of UK 1st class cricketers born prior to 1840*

Description

Year and birth, lifespan, etc, of British first class cricketers, born prior to 1840, whose handedness could be determined.

Usage

```
data(earlycrcktr)
```

Format

A data frame with 211 observations on the following 8 variables.

- `left` a factor with levels `right` `left`
- `year` numeric, year of birth
- `life` numeric, lifespan
- `dead` numeric (all 1 = dead)
- `acd` numeric (0 = not accidental or not dead, 1 = accidental death)
- `kia` numeric (all 0 = not killed in action)
- `inbed` numeric (0 = did not die in bed, 1 = died in bed)
- `cause` a factor with levels `alive` `acd` (accidental death) `inbed` (died in bed)

Source

John Aggleton, Martin Bland. Data were collated as described in Aggleton et al.

References

Aggleton JP, Bland JM, Kentridge RW, Neave NJ 1994. Handedness and longevity: an archival study of cricketers. British Medical Journal 309, 1681-1684.

Bailey P, Thorne P, Wynne-Thomas P. 1993. Who's Who of Cricketers. 2nd ed, London, Hamlyn.

Bland M and Altman D. 2005. Do the left-handed die young? Significance 2, 166-170.

See Also

`cricketer`.

Examples

```
data(earlycrcktr)
```

`fishRivers` *Characteristics of river reference sites*

Description

Data on river sites was matched with data on fish caught at nearby fishing sites. Fishing sites were classified according to type of fish (`Group` = fish river type).

Usage

```
data(fishRivers)
```

Format

A data frame with 128 observations on the following 23 variables.

`Group` Fish river types; a factor with levels `F1` `F2` `F3` `F4` `F5` `F6`

`Alk` mgm/l of CaCO3

`Avrain` mean annual rainfall, mm

`Bedrock` % cover on river edge attached to substratum

`Boulder` numeric, % cover of stones $>$ 200mm diameter

`Cobble` numeric, % cover of stones between 60 \& 200mm

`Cond` electrical conductivity (muS/cm)

`Dis` maximum distance from source (m)

`DO` numeric, dissolved O2, % saturation

`Elev` nearest contour line (m) below site

`Fine` numeric, % cover of particles $<$ 0.02mm

`Gravel` numeric, % cover of particles between 2 \& 20mm

`Lat` latitude

`Long` longitude

`NoxN` total oxidised Nitrogen (mgl/l)

`Pebble` numeric, % cover of stones between 20 \& 60mm

`Ph` a numeric vector

`Sand` numeric, % cover of particles between 0.02 \& 2mm

`Slope` numeric, elevatioon difference (m) between site \& a point 1km upstream

`Tkn` total Kjeldahl Nitrogen (mgl/l)

`Tpl` total Phosphorus (mgl/l)

`Turb` turbidity (NTU)

`Width` modal river width (m), assessed visually

Details

To what extent can the fish river type be predicted, based on: (i) all explanatory variables; (ii) the variables `Avrain`, `Dis`, `Elev`, `Lat`, `Long` and `Slope`. The second set comprises the variables that would be used in practice to predict the fish river type at other sites.

Source

Data relate to Turak and Koop (2007).

References

Turak, E and Koop, K. Multi-attribute ecological river typology for assessing ecological condition and conservation planning. Hydrobiologica 603:83-104, 2007.

Examples

```
data(fishRivers)
library(MASS)
fish.lda <- lda(Group ~ Avrain + Dis +Elev + Lat + Long + Slope,
                data=fishRivers)
```

`fumig` *Profiles of fumigant concentration over time*

Description

Fumigant concentrations are given at six times through a 120 minute fumigation, for seven different runs of the fumigation procedure.

Usage

```
fumig
```

Format

A data frame with 8 observations on the following 8 variables.

`testnam` a factor with levels `Applied Test 1` `Applied Test 2` `Applied Test 3` `Confirmation Test Query` `Applied Test 4` `Applied Test` `Applied Test 5` `Applied Test 6`

`Cultivar` a factor with levels `Bogapple` `Chewton Pear` (the names are invented)

`X1` concentration (gm/cm^3) at 5 minutes

`X2` concentration at 10 min

`X3` concentration at 30 min

`X4` concentration at 60 min

`X5` concentration at 90 min

`X6` concentration at 120 min

Details

Sortpion of fumigant by the fruit, different between different cultivars, is the main reason for the decline in concentration over time.

Source

John Maindonald

References

For s discussion of the technology, see:

Maindonald, J.H.; Waddell B.C.; Birtles D.B. 1991. Response to Methyl Bromide Fumigation of Codling Moth (Lepidoptera: Tortricidae) on Cherries. *Journal of Economic Entomology* 85: 1222-1230.

Examples

```
data(fumig)
```

MaskedPriming *Masked Repetition Priming Data*

Description

Words (`words`), preceded by an invisible identical or unrelated “prime”, or non-words, were flashed in front of subjects (`id`). The time taken by the subject to identify the letter combination as “not word” or “word” was then measured.

Usage

```
data(MaskedPriming)
```

Format

A data frame with 6381 correct responses to words on the following 10 variables.

`subjects` a factor with levels `1` to `72`

`words` a factor with levels `1` to `192`

`e` the level of familiarity, a factor with levels `1` `2` `3`

`ct` a factor with levels `HI` `HU` `LI` `LU`. Here, `HI` = high freq, identical prime; HU = high freq, unrelated prime; `LI` = low, identical; LU = low, unrelated

`f` the word frequency, a numeric vector with values -0.5 (High) and 0.5 (Low)

`p` priming, a numeric vector with values -0.5 (Identical word) and 0.5 (Unrelated word)

`rt` reaction time (milliseconds), a numeric vector

`srt` reaction time (sec) = `rt`/1000, a numeric vector

`lrt` loge(reaction time), a numeric vector

`rrt` negative of speed of reaction = -1/`srt`, a numeric vector

Details

This combines the datasets from Bodner and Masson (1997, Exp 1 and Exp 2a) and Kinoshita (2006, Exp 2).

Source

Kliegl et al (2008)

References

Bodner, G.E., and Masson, M. E. J. 1997 Masked repetition priming of words and nonwords: Evidence for a nonlexical basis for priming. *Journal of Memory and Language* **37**, 268-293.

Kinoshita, S. 2006 Additive and interactive effects of word frequency and masked repetition in the lexical decision task. *Psychonomic Bulletin & Review* **13**, 668-673.

Kliegl, R., Masson, M. E. J. and Richter, E. M. 2008. A linear mixed-effects model analysis of masked repetition priming. *Manuscript*.

Examples

```
data(MaskedPriming)
str(MaskedPriming)
plot(MaskedPriming[sample(6381,100), 7:10])
## Not run:
library(lme4)
cmat <- matrix(c(-1,  1,  0,
            -1, -1,  2),  3,  2,
            dimnames=list(c("BM1", "BM2", "SK"),
                          c(".BM1-2", ".BM-SK")))
m0 <- lmer(rrt ~ p*f*e + (1 | subjects) + (0 + p | subjects) +
           (0 + f | subjects) + (1 | words), contrasts=list(e=cmat),
           data=d)
```

```
m1p <- lmer(rrt ~ p*f*e + (p | subjects) + (0+f | subjects) + (1 | words),
                 contrasts=list(e=cmat)
m2 <- lmer(rrt ~ p*f*e + (p + f | subjects) + (1 | words),
                 contrasts=list(e=cmat), data=d)
anova(m0, m1p, m2)

## End(Not run)
```

---

plotSampDist *Plot(s) of simulated sampling distributions*

Description

Plots are based on the output from `simulateSampDist()`. By default, both density plots and normal probability plots are given, for a sample from the specified population and for samples of the relevant size(s)

Usage

```
plotSampDist(sampvalues, graph = c("density", "qq"), cex = 0.925,
             titletext = "Empirical sampling distributions of the",
             popsample=TRUE, ...)
```

Arguments

`sampvalues`	Object output from `simulateSampDist()`
`graph`	Either or both of `"density"` and `"qq"`
`cex`	Character size parameter, relative to default
`titletext`	Title for graph
`popsample`	If `TRUE` show distribution of random sample from population
`...`	Other graphics parameters

Value

Plots graph(s), as described above.

Author(s)

John Maindonald

References

Maindonald, J.H. and Braun, W.J. (3rd edn, 2010) “Data Analysis and Graphics Using R”, Sections 3.3 and 3.4.

See Also

See Also `help(simulateSampDist)`

Examples

```
## By default, sample from normal population
simAvs <- simulateSampDist()
par(pty="s")
plotSampDist(simAvs)
## Sample from empirical distribution
simAvs <- simulateSampDist(rpop=rivers)
plotSampDist(simAvs)

## The function is currently defined as
function(sampvalues, graph=c("density", "qq"), cex=0.925,
           titletext="Empirical sampling distributions of the",
           popsample=TRUE, ...){
    if(length(graph)==2)oldpar <- par(mfrow=c(1,2), mar=c(3.1,4.1,1.6,0.6),
               mgp=c(2.5, 0.75, 0), oma=c(0,0,1.5,0), cex=cex)
    values <- sampvalues$values
    numINsamp <- sampvalues$numINsamp
    funtxt <- sampvalues$FUN
    nDists <- length(numINsamp)+1
    nfirst <- 2
    legitems <- paste("Size", numINsamp)
    if(popsample){nfirst <- 1
                  legitems <- c("Size 1", legitems)
                }
    if(match("density", graph)){
      popdens <- density(values[,1], ...)
      avdens <- vector("list", length=nDists)
      maxht <- max(popdens$y)
      ## For each sample size specified in numINsamp, calculate mean
      ## (or other statistic specified by FUN) for numsamp samples
      for(j in nfirst:nDists){
        av <- values[, j]
        avdens[[j]] <- density(av, ...)
        maxht <- max(maxht, avdens[[j]]$y)
      }
    }
    if(length(graph)>0)
      for(graphtype in graph){
        if(graphtype=="density"){
          if(popsample)
          plot(popdens, ylim=c(0, 1.2*maxht), type="l", yaxs="i",
               main="")
          else plot(avdens[[2]], type="n", ylim=c(0, 1.2*maxht),
                    yaxs="i", main="")
          for(j in 2:nDists)lines(avdens[[j]], col=j)
          legend("topleft",
                 legend=legitems,
                 col=nfirst:nDists, lty=rep(1,nDists-nfirst+1), cex=cex)
        }
        if(graphtype=="qq"){
          if(popsample) qqnorm(values[,1], main="")
          else qqnorm(values[,2], type="n")
```

```
          for(j in 2:nDists){
            qqav <- qqnorm(values[, j], plot.it=FALSE)
            points(qqav, col=j, pch=j)
           }
            legend("topleft", legend=legitems,
                   col=nfirst:nDists, pch=nfirst:nDists, cex=cex)
       }
      }
    if(par()$oma[3]>0){
      outer <- TRUE
      line=0
    }  else
    {
      outer <- FALSE
      line <- 1.25
    }
    if(!is.null(titletext))
      mtext(side=3, line=line,
            paste(titletext, funtxt),
            cex=1.1, outer=outer)
    if(length(graph)>1)par(oldpar)
  }
```

---

simulateSampDist *Simulated sampling distribution of mean or other statistic*

Description

Simulates the sample distribution of the specified statistic, for samples of the size(s) specified in numINsamp. Additionally a with replacement) sample is drawn from the specified population.

Usage

```
simulateSampDist(rpop = rnorm, numsamp = 100, numINsamp = c(4, 16),
                   FUN = mean, seed=NULL
       )
```

Arguments

rpop	Either a function that generates random samples from the specified distribution, or a vector of values that define the population (i.e., an empirical distribution)
numsamp	Number of samples that should be taken. For close approximation of the asymptotic distribution (e.g., for the mean) this number should be large
numINsamp	Size(s) of each of the numsamp sample(s)
FUN	Function to calculate the statistic whose sampling distribution is to be simulated
seed	Optional seed for random number generation

Value

List, with elements `values`, `numINsamp` and `FUN`

values
: Matrix, with dimensions `numsamp` by `numINsamp + 1`. The first column has a random with replacement sample from the population, while the remaining `length(numINsamp)` columns hold simulated values from sampling distributions with samples of the specified size(s)

numINsamp
: Input value of `numINsamp`

numsamp
: Input value of `numsamp`

Author(s)

John Maindonald

References

Maindonald, J.H. and Braun, W.J. (2nd edn, 2006) *Data Analysis and Graphics Using R*, 2nd edn, Section 4.1

See Also

`help(plotSampDist)`

Examples

```
## By default, sample from normal population
simAvs <- simulateSampDist()
par(pty="s")
plotSampDist(simAvs)
## Sample from empirical distribution
simAvs <- simulateSampDist(rpop=rivers)
plotSampDist(simAvs)


## The function is currently defined as
function(rpop=rnorm, numsamp=100, numINsamp=c(4,16), FUN=mean,
seed=NULL){
    if(!is.null(seed))set.seed(seed)
    funtxt <- deparse(substitute(FUN))
    nDists <- length(numINsamp)+1
    values <- matrix(0, nrow=numsamp, ncol=nDists)
    if(!is.function(rpop)) {
      x <- rpop
      rpop <- function(n)sample(x, n, replace=TRUE)
    }
    values[,1] <- rpop(numsamp)
    for(j in 2:nDists){
      n <- numINsamp[j-1]
      for(i in 1:numsamp)values[i, j] <- FUN(rpop(n))
    }
    colnames(values) <- paste("Size", c(1, numINsamp))
```

```
  invisible(list(values=values, numINsamp=numINsamp, FUN=funtxt))
}
```

SOI *Southern Oscillation Index Data*

Description

The Southern Oscillation Index (SOI) is the difference in barometric pressure at sea level between Tahiti and Darwin. Monthly and annual SOI data, for the years 1876-2011, are given.

Usage

```
SOI
```

Format

This data frame contains the following columns:

Year a numeric vector

Jan average January SOI values for each year

Feb average February SOI values for each year

Mar average March SOI values for each year

Apr average April SOI values for each year

May average May SOI values for each year

Jun average June SOI values for each year

Jul average July SOI values for each year

Aug average August SOI values for each year

Sep average September SOI values for each year

Oct average October SOI values for each year

Nov average November SOI values for each year

Dec average December SOI values for each year

avsoi average annual SOI values

Source

Australian Bureau of Meteorology web pages:

http://www.bom.gov.au/climate/current/soihtm1.shtml

References

Nicholls, N., Lavery, B., Frederiksen, C.\ and Drosdowsky, W. 1996. Recent apparent changes in relationships between the El Nino – southern oscillation and Australian rainfall and temperature. Geophysical Research Letters 23: 3357-3360.

Examples

```
plot(ts(SOI[, "avsoi"], start=1900),
     panel=function(y,...)panel.smooth(1900:2008, y,...))
```

Index