Package ‘fastR’

December 18, 2015

Type Package

Title Foundations and Applications of Statistics Using R

Version 0.10.2

Date 2015-12-17

Author Randall Pruim

Maintainer Randall Pruim <rpruim@calvin.edu>

Description Data sets and utilities to accompany
``Foundations and Applications of Statistics: an Introduction using R'' (R Pruim, published by AMS, 2011), a text covering topics from probability and mathematical statistics at an advanced undergraduate level. R is integrated throughout, and access to all the R code in the book is provided via the snippet function.

License GPL (>= 2)

Depends R (>= 3.0.0), mosaic

Imports lattice, grid, mosaicData

Suggests MASS, faraway, Hmisc, DAAG, multcomp, vcd, car, alr3, corrgram, BradleyTerry2, cubature, Devore7, knitr

VignetteBuilder knitr

LazyLoad yes

LazyData yes

RoxygenNote 5.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2015-12-18 06:43:12

R topics documented:

fastR-package 3
actgpa 4
airlineArrival 4

airpollution 5
balldrop 5
batting 6
births78 7
buckthorn 8
bugs 8
col.fastR 9
col.perc 10
concrete 10
corn 11
cuckoo 12
deathPenalty 13
drag 13
endurance 14
familySmoking 15
fumbles 16
geolm 17
givenOrder 18
golfballs 18
gpa 19
heliumFootballs 19
ice 20
inflation 22
Jordan8687 22
kids 23
littleSurvey 24
mathnoise 25
miaa05 26
mlb2004 27
ncaa2010 28
nfl2007 29
nlmax 29
noise 30
palettes 31
paperplanes 32
pendulum 33
petstress 34
pheno 35
pigs 36
pitching2005 37
poison 39
punting 40
ratpoison 41
rgolfballs 41
rubberband 42
scent 42
snippet 43
soap 44

spheres . . . 45
step . . . 45
stereogram . . . 46
students . . . 47
tastetest . . . 48
tdf . . . 49
tirewear . . . 49
traffic . . . 50
trebuchet . . . 51
undocumented . . . 52
utilities . . . 52
vaov . . . 53
wilson.ci . . . 54
workingWomen . . . 55
xplot . . . 56

Index **58**

`fastR-package` *Foundations and Applications of Statistics*

Description

Data sets and utility functions to accompany *Foundations and Applications of Statistics: An Introduction Using R* by Randall Pruim.

Author(s)

Randall Pruim

Maintainer: Randall Pruim <rpruim@calvin.edu>

References

R. Pruim, *Foundations and Applicaitons of Statistics: An Introduction Using R*, AMS, 2011.

Examples

```
require(fastR)
trellis.par.set(theme=col.fastR())
```

`actgpa` *ACT scores and GPA*

Description

ACT scores and college GPA for a small sample of college students.

Format

A data frame with 26 observations on the following 2 variables.

- ACT a numeric vector
- GPA a numeric vector

Examples

```
if (require(mosaicData)) {
  xyplot(GPA ~ ACT, data=actgpa)
}
```

`airlineArrival` *Airline On-Time Arrival Data*

Description

Flights categorized by destination city, airline, and whether or not the flight was on time.

Format

A data frame with 11000 observations on the following 3 variables.

- Airport a factor with levels `LosAngeles` `Phoenix` `SanDiego` `SanFrancisco` `Seattle`
- Result a factor with levels `Delayed` `OnTime`
- Airline a factor with levels `Alaska` `AmericaWest`

Source

Barnett, Arnold. 1994. "How numbers can trick you." *Technology Review*, vol. 97, no. 7, pp. 38–45.

References

These and similar data appear in many text books under the topic of Simpson's paradox.

Examples

```
row.perc(xtabs(~Airline+Result, data=airlineArrival))
for (city in levels(airlineArrival$Airport)) {
cat(paste('\nArriving in ', city,':\n',sep=''))
print(row.perc(xtabs(~Airline+Result, airlineArrival,
subset=Airport==city)))
}
```

`airpollution` *Air pollution measurements*

Description

Air pollution measurements at three locations.

Format

A data frame with 6 observations on the following 2 variables.

- pollution a numeric vector
- location a factor with levels `Hill Suburb` `Plains Suburb` `Urban City`

Source

David J. Saville and Graham R. Wood, *Statistical methods: A geometric primer*, Springer, 1996.

Examples

```
data(airpollution)
summary(lm(pollution ~ location, data=airpollution))
```

`balldrop` *Ball dropping data*

Description

Undergraduate students in a physics lab recorded the height from which a ball was dropped and the time it took to reach the floor.

Format

A data frame with 30 observations on the following 2 variables.

- height height in meters
- time time in seconds

Source

Steve Plath, Calvin College Physics Department

Examples

```
xyplot(time ~ height, data=balldrop)
```

`batting` *Major League Batting 2000-2005*

Description

Major League batting data for the seasons from 2000-2005.

Format

A data frame with 8062 observations on the following 22 variables.

- player unique identifier for each player
- year year
- stint for players who were traded mid-season, indicates which portion of the season the data cover
- team three-letter code for team
- league a factor with levels `AA` `AL` `NL`
- G games
- AB at bats
- R runs
- H hits
- H2B doubles
- H3B triples
- HR home runs
- RBI runs batted in
- SB stolen bases
- CS caught stealing

- BB bases on balls (walks)
- SO strike outs
- IBB intentional base on balls
- HBP hit by pitch
- SH a numeric vector
- SF sacrifice fly
- GIDP grounded into double play

Examples

```
data(batting)
histogram(~HR,batting)
```

---

births78 *Births by Day*

Description

The number of live births in the United States for each day of the year 1978.

Format

A data frame with 365 observations on the following 3 variables.

- date date (as a factor)
- births number of live births
- dayofyear number of days since start of year

Examples

```
data(births78)
xyplot(births ~ dayofyear, births78)
```

buckthorn — *Buckthorn*

Description

Data from an experiment to determine the efficacy of various methods of eradicating buckthorn, an invasive woody shrub. Buckthorn plants were chopped down and the stumps treated with various concentrations of glyphosate. The next season, researchers returned to see whether the plant had regrown.

Format

A data frame with 165 observations on the following 3 variables.

- shoots number of new shoots coming from stump
- conc concentration of glyphosate applied
- dead weather the stump was considered dead

Source

David Dornbos, Calvin College

Examples

```
data(buckthorn)
```

bugs — *Bugs*

Description

This data frame contains data from an experiment to see if insects are more attracted to some colors than to others. The researchers prepared colored cards with a sticky substance so that insects that landed on them could not escape. The cards were placed in a field of oats in July. Later the researchers returned, collected the cards, and counted the number of cereal leaf beetles trapped on each card.

Format

A data frame with 24 observations on the following 2 variables.

- Color color of card; one of `B`(lue) `G`(reen) `W`(hite) `Y`(ellow)
- NumTrap number of insects trapped on the card

Source

M. C. Wilson and R. E. Shade, Relative attractiveness of various luminescent colors to the cereal leaf beetle and the meadow spittlebug, *Journal of Economic Entomology* 60 (1967), 578–580.

Examples

```
data(bugs)
summary(NumTrap ~ Color, bugs, fun=favstats)
```

---

col.fastR *Lattice Theme*

Description

A theme for use with lattice graphics.

Usage

```
col.fastR(bw = FALSE, lty = 1:7)
```

Arguments

`bw`	whether color scheme should be "black and white"
`lty`	vector of line type codes

Value

Returns a list that can be supplied as the `theme` to `trellis.par.set()`.

Note

This theme was used in the production of the book *Foundations and Applications of Statistics*

Author(s)

Randall Pruim

See Also

`trellis.par.set`, `show.settings`

Examples

```
trellis.par.set(theme=col.fastR(bw=TRUE))
show.settings()
trellis.par.set(theme=col.fastR())
show.settings()
```

col.perc *Row and Column Percentages*

Description

Convenience wrappers around `apply()` to compute row and column percentages of matrix-like structures, including output of `xtabs`.

Usage

```
col.perc(x)

row.perc(x)
```

Arguments

x	matrix-like structure

Author(s)

Randall Pruim

Examples

```
row.perc(xtabs(~Airline+Result,airlineArrival))
```

concrete *Concrete Compressive Strength Data*

Description

These data were collected by I-Cheng Yeh to determine how the compressive strength of concrete is related to its ingredients (cement, blast furnace slag, fly ash, water, superplasticizer, coarse aggregate, and fine aggregate) and age.

Format

`concreteAll` is a data frame with the following 9 variables.

- cement amount of cement (kg/m^3)
- slag amount of blast furnace slag (kg/m^3)
- ash amount of fly ash(kg/m^3)
- water amount of water (kg/m^3)

- superP amount of superplasticizer (kg/m^3)
- coarseAg amount of coarse aggregate (kg/m^3)
- fineAg amount of fine aggregate (kg/m^3)
- age age of concrete in days
- strength compressive strength measured in MPa

`concrete28` is a subset of `concreteAll`.

Source

Data were obtained from the Machine Learning Repository (http://archive.ics.uci.edu/ml/) where they were deposited by I-Cheng Yeh (`<icyeh at chu.edu.tw>`) who retains the copyright for these data.

References

I-Cheng Yeh (1998), "Modeling of strength of high performance concrete using artificial neural networks," *Cement and Concrete Research*, Vol. 28, No. 12, pp. 1797-1808.

Examples

```
data(concreteAll)
data(concrete28)
```

`corn` *Corn Yield*

Description

William Gosset analyzed data from an experiment comparing the yield of regular and kiln-dried corn.

Format

A data frame with 11 observations on the following 2 variables.

- reg yield of regular corn (lbs/acre)
- kiln yield of kiln-dried corn (lbs/acre)

Details

Gosset (Student) reported on the results of seeding plots with two different kinds of seed. Each type of seed (regular and kiln-dried) was planted in adjacent plots, accounting for 11 pairs of "split" plots.

Source

These data are also available at DASL, the data and story library (http://lib.stat.cmu.edu/DASL/).

References

W.S. Gosset, "The Probable Error of a Mean," Biometrika, 6 (1908), pp 1-25.

Examples

```
corn2 <- stack(corn)
names(corn2) <- c('yield','treatment')
lm(yield ~ treatment, data = corn2)
t.test(yield ~ treatment, data=corn2)
t.test(corn$reg, corn$kiln)
```

---

cuckoo *Cuckoo eggs in other birds' nests*

Description

Cuckoos are knows to lay their eggs in the nests of other (host) birds. The eggs are then adopted and hatched by the host birds. These data were originally collected by O. M. Latter in 1902 to see how the size of a cuckoo egg is related to the species of the host bird.

Format

A data frame with 120 observations on the following 2 variables.

- length length of egg (mm)
- species a factor with levels `hedge sparrow` `meadow pipet` `pied wagtail` `robin` `tree pipet` `wren`

Source

L.H.C. Tippett, *The Methods of Statistics*, 4th Edition, John Wiley and Sons, Inc., 1952, p. 176.

References

These data are also available from DASL, the data and story library (http://lib.stat.cmu.edu/DASL/).

Examples

```
data(cuckoo)
bwplot(length~species,cuckoo)
```

deathPenalty — *Death Penalty and Race*

Description

A famous example of Simpson's paradox.

Format

A data frame with 326 observations. The factors are coded more succinctly in `deathPen`, but otherwise the data are the same.

- DeathPenalty a factor with levels `Yes` `No`
- Penalty a factor with levels `Death` `Not`
- Victim a factor with levels `Black` `White` (or `Bl` `Wh`)
- Defendant a factor with levels `Black` `White` (or `Bl` `Wh`)

Source

Radelet, M. (1981). Racial characteristics and imposition of the death penalty. *American Sociological Review*, 46:918–927.

Examples

```
xtabs(~Defendant+Penalty,deathPenalty)
xtabs(~Defendant+Victim+Penalty,deathPenalty)
```

drag — *Drag force experiment*

Description

The data come from an experiment to determine how terminal velocity depends on the mass of the falling object. A helium balloon was rigged with a small basket and just the ballast to make it neutrally buoyant. Mass was then added and the terminal velocity is calculated by measuring the time it took to fall between two sensors once terminal velocity has been reached. Larger masses were drop from higher heights and used sensors more widely spaced.

Format

A data frame with 42 observations on the following 5 variables.

- time time (in seconds) to travel between two sensors
- mass net mass (in kg) of falling object
- height distance (in meters) between two sensors
- velocity average velocity (in m/s) computed from `time` and `height`
- force.drag calculated drag force (in N, `force.drag = mass * 9.8`) using the fact that at terminal velocity, the drag force is equal to the force of gravity

Source

Calvin College physics students under the supervision of Professor Steve Plath.

Examples

```
data(drag)
with(drag,force.drag / mass)
xyplot(velocity ~ mass, drag)
```

---

endurance *Endurance and vitamin C*

Description

The effect of a single 600 mg dose of ascorbic acid versus a sugar placebo on the muscular endurance (as measured by repetitive grip strength trials) of fifteen male volunteers (19-23 years old).

Format

A data frame with 15 observations on the following 5 variables.

- Vitamin number of repetitions until reaching 50 maximal grip after taking viatimin
- First which treatment was done first, a factor with levels `Placebo` `Vitamin`
- Placebo number of repetitions until reaching 50 strength after taking placebo

Details

Three initial maximal contractions were performed for each subject, with the greatest value indicating maximal grip strength. Muscular endurance was measured by having the subjects squeeze the dynamometer, hold the contraction for three seconds, and repeat continuously until a value of 50 maximum grip strength was achieved for three consecutive contractions. Endurance was defined as the number of repetitions required to go from maximum grip strength to the initial 50 positive verbal encouragement in an effort to have them complete as many repetitions as possible.

The study was conducted in a double-blind manner with crossover.

Source

These data are available from OzDASL, the Australasian data and story library (http://www.statsci.org/data/).

References

Keith, R. E., and Merrill, E. (1983). The effects of vitamin C on maximum grip strength and muscular endurance. *Journal of Sports Medicine and Physical Fitness*, 23, 253-256.

Examples

```
data(endurance)
t.test(endurance$Vitamin, endurance$Placebo, paired=TRUE)
t.test(log(endurance$Vitamin), log(endurance$Placebo), paired=TRUE)
t.test(1/endurance$Vitamin, 1/endurance$Placebo, paired=TRUE)
xqqmath( ~ Vitamin - Placebo, data = endurance)
xqqmath( ~ log(Vitamin) - log(Placebo), data = endurance)
xqqmath( ~ 1/Vitamin - 1/Placebo, data = endurance)
```

`familySmoking` *Family smoking*

Description

A cross-tabulation of whether a student smokes and how many of his or her parents smoke from a study conducted in the 1960's.

Format

A data frame with 5375 observations on the following 2 variables.

- Student a factor with levels `DoesNotSmoke` `Smokes`
- Parents a factor with levels `NeitherSmokes` `OneSmokes` `BothSmoke`

Source

S. V. Zagona (ed.), *Studies and issues in smoking behavior*, University of Arizona Press, 1967.

References

The data also appear in

Brigitte Baldi and David S. Moore, *The Practice of Statistics in the Life Sciences*, Freeman, 2009.

Examples

```
data(familySmoking)
xchisq.test( xtabs(~Parents + Student, familySmoking) )
```

---

fumbles *NCAA football fumbles*

Description

This data frame gives the number of fumbles by each NCAA FBS team for the first three weeks in November, 2010.

Format

A data frame with 120 observations on the following 7 variables.

- team NCAA football team
- rank rank based on fumbles per game through games on November 26, 2010
- W number of wins through games on November 26, 2010
- L number of losses through games on November 26, 2010
- week1 number of fumbles on November 6, 2010
- week2 number of fumbles on November 13, 2010
- week3 number of fumbles on November 20, 2010

Details

The fumble counts listed here are total fumbles, not fumbles lost. Some of these fumbles were recovered by the team that fumbled.

Source

http://www.teamrankings.com/college-football/stat/fumbles-per-game

Examples

```
data(fumbles)
m <- max(fumbles$week1)
table(factor(fumbles$week1,levels=0:m))
favstats( ~ week1, data=fumbles)
# compare with Poisson distribution
signif( cbind(
  fumbles=0:m,
  observedCount=table(factor(fumbles$week1,levels=0:m)),
  modelCount= 120* dpois(0:m,mean(fumbles$week1)),
```

```
    observedPct=table(factor(fumbles$week1,levels=0:m))/120,
    modelPct= dpois(0:m,mean(fumbles$week1))
  ) ,3)
  showFumbles <- function(x,lambda=mean(x),...) {
  mx <- max(x)
      result <- histogram(~x, type="density", xlim=c(-.5,(mx+2.5)),
  xlab='number of fumbles',
          panel=function(x,y,...){
              panel.histogram(x,alpha=0.8,breaks=seq(-0.5,(mx+2.5),by=1,...))
              panel.points(0:(mx+2),dpois(0:(mx+2),lambda),pch=19,alpha=0.8)
          }
      )
      print(result)
      return(result)
  }
  showFumbles(fumbles$week1)
  showFumbles(fumbles$week2)
  showFumbles(fumbles$week3)
```

---

geolm *Geometric representation of linear model*

Description

geolm create a graphical representation of the fit of a linear model.

Usage

```
geolm(formula, data = parent.env(), type = "xz", version = 1,
  plot = TRUE, ...)

to2d(x, y, z, type = NULL, xas = c(0.4, -0.3), yas = c(1, 0), zas = c(0,
  1))
```

Arguments

formula	a formula as used in lm.
data	a data frame as in lm.
type	character: indicating the type of projection to use to collapse multi-dimensional data space into two dimensions of the display.
version	an integer (currently 1 or 2). Which version of the plot to display.
plot	a logical: should the plot be displayed?
x, y, z	numeric.
xas, yas, zas	numeric vector of length 2 indicating the projection of c(1,0,0), c(0,1,0), and c(0,0,1).
...	other arguments passed to lm

Author(s)

Randall Pruim

See Also

lm.

Examples

```
geolm(pollution ~ location, data=airpollution)
geolm(distance ~ projectileWt, data=trebuchet2)
```

---

givenOrder *Create ordered factor with order inferred from order given*

Description

The order of the resulting factor is determined by the order in which unique labels first appear in the vector or factor x.

Usage

```
givenOrder(x)
```

Arguments

x	a vector or factor to be converted into an ordered factor.

Examples

```
givenOrder(c("First", "Second", "Third", "Fourth", "Fifth", "Sixth"))
```

---

golfballs *Golf ball numbers*

Description

Allan Rossman used to live on a golf course in a spot where dozens of balls would come into his yard every week. He collected the balls and eventually tallied up the numbers on the first 5000 golf balls he collected. Of these 486 bore the number 1, 2, 3, or 4. The remaining 14 golf balls were omitted from the data.

Format

The format is: num [1:4] 137 138 107 104

Source

Data collected by Allan Rossman in Carlisle, PA.

Examples

```
data(golfballs)
golfballs/sum(golfballs)
chisq.test(golfballs, p=rep(.25,4))
```

---

gpa *GPA, ACT, and SAT scores*

Description

GPA, ACT, and SAT scores for a sample of students.

Format

A data frame with 271 observations on the following 4 variables.

- satm SAT mathematics score
- satv SAT verbal score
- act ACT score
- gpa college grade point average

Examples

```
data(gpa)
splom(gpa)
```

---

heliumFootballs *Punting helium- and air-filled footballs*

Description

Two identical footballs, one air-filled and one helium-filled, were used outdoors on a windless day at The Ohio State University's athletic complex. Each football was kicked 39 times and the two footballs were alternated with each kick. The experimenter recorded the distance traveled by each ball.

Format

A data frame with 39 observations on the following 3 variables.

- Trial trial number
- Air distance traveled by air-filled football (yards)
- Helium distance traveled by helium-filled football (yards)

Source

These data are available from DASL, the data and story library (http://lib.stat.cmu.edu/DASL/).

References

Lafferty, M. B. (1993), "OSU scientists get a kick out of sports controversy", *The Columbus Dispatch* (November, 21, 1993), B7.

Examples

```
data(heliumFootballs)
xyplot(Helium~Air, data=heliumFootballs)
bwplot(~(Helium-Air), data=heliumFootballs)
```

---

ice *Cooling muscles with ice*

Description

This data set contains the results of an experiment comparing the efficacy of different forms of dry ice application in reducing the temperature of the calf muscle.

Details

The 12 subjects in this study came three times, at least four days apart, and received one of three ice treatments (cubed ice, crushed ice, or ice mixed with water). In each case, the ice was prepared in a plastic bag and applied dry to the subjects calf muscle. The temperature measurements were taken on the skin surface and inside the calf muscle (via a 4 cm long probe) every 30 seconds for 20 minutes prior to icing, for 20 minutes during icing, and for 2 hours after the ice had been removed. The temperature measurements are stored in variables that begin with `B` (baseline), `T` (treatment), or `R` (recovery) followed by a numerical code for the elapsed time formed by concatenating the number of minutes and seconds. For example, `R1230` contains the temperatures 12 minutes and 30 seconds after the ice had been removed.

Variables include

- Subject identification number

- Sex a factor with levels `female` `male`
- Weight weight of subject (kg)
- Height height of subject (cm)
- Skinfold skinfold thickness
- Calf calf diameter (cm)
- Age age of subject
- Location a factor with levels `intramuscular` `surface`
- Treatment a factor with levels `crushed` `cubed` `wet`
- B0 baseline temperature at time 0
- B30 baseline temperature 30 seconds after start
- B100 baseline temperature 1 minute after start
- B1930 baseline temperature 19 minutes 30 seconds start
- T0 treatment temperature at beginning of treatment
- T30 treatment temperature 30 seconds after start of treatment
- T100 treatment temperature 1 minute after start of treatment
- T1930 treatment temperature 19 minutes 30 seconds after start of treatment
- R0 recovery temperature at start of recovery
- R30 recovery temperature 30 seconds after start of recovery
- R100 recovery temperature 1 minute after start of recovery
- R12000 recovery temperature 120 minutes after start of recovery

Source

Dykstra, J. H., Hill, H. M., Miller, M. G., Michael T. J., Cheatham, C. C., and Baker, R.J., Comparisons of cubed ice, crushed ice, and wetted ice on intramuscular and surface temperature changes, *Journal of Athletic Training* 44 (2009), no. 2, 136–141.

Examples

```
data(ice)
xyplot(Weight ~ Skinfold, groups=Sex, data=ice, auto.key=TRUE)
```

inflation *Inflation data*

Description

The article developed four measures of central bank independence and explored their relation to inflation outcomes in developed and developing countries. This datafile deals with two of these measures in 23 nations.

Format

A data frame with 23 observations on the following 5 variables.

- country country where data were collected
- ques questionnaire index of independence
- inf annual inflation rate, 1980-1989 (percent)
- legal legal index of independence
- dev developed (1) or developing (2) nation

Source

These data are available from OzDASL, the Australasian Data and Story Library (http://www.statsci.org/data/).

References

A. Cukierman, S.B. Webb, and B. Negapi, "Measuring the Independence of Central Banks and Its Effect on Policy Outcomes," World Bank Economic Review, Vol. 6 No. 3 (Sept 1992), 353-398.

Examples

```
data(inflation)
```

Jordan8687 *Michael Jordan personal scoring*

Description

The number of points scored by Michael Jordan in each game of the 1986-87 regular season.

Format

A data frame with 82 observations on the following 2 variables.

- Gamea numeric vector
- Pointsa numeric vector

Examples

```
if (require(mosaicData)) {
  data(Jordan8687)
  xqqmath(~Points, data=Jordan8687)
}
```

---

kids *Goals and popularity factors for school kids*

Description

Subjects were students in grades 4-6 from three school districts in Michigan. Students were selected from urban, suburban, and rural school districts with approximately 1/3 of their sample coming from each district. Students indicated whether good grades, athletic ability, or popularity was most important to them. They also ranked four factors: grades, sports, looks, and money, in order of their importance for popularity. The questionnaire also asked for gender, grade level, and other demographic information.

Format

A data frame with 478 observations on the following 11 variables.

- Gender a factor with levels boy `girl`
- Grade grade in school
- Age student age
- Race a factor with levels `Other` `White`
- Urban.Rural a factor with levels `Rural` `Suburban` `Urban`
- School a factor with levels `Brentwood Elementary` `Brentwood Middle` `Brown Middle` `Elm` `Main` `Portage` `Ridge` `Sand` `Westdale Middle`
- Goals a factor with levels `Grades` `Popular` `Sports`
- Grades rank of ‘make good grades’ (1=most important for popularity; 4=least important)
- Sports rank of ‘beging good at sports’ (1=most important for popularity; 4=least important)
- Looks rank of ‘beging handsome or pretty’ (1=most important for popularity; 4=least important)
- Money rank of ‘having lots of money’ (1=most important for popularity; 4=least important)

Source

These data are available at DASL, the data and story library (`http://lib.stat.cmu.edu/DASL/`).

References

Chase, M. A., and Dummer, G. M. (1992), "The Role of Sports as a Social Determinant for Children," Research Quarterly for Exercise and Sport, 63, 418-424.

Examples

```
data(kids)
xtabs(~Goals + Urban.Rural, data=kids)
chisq.test(xtabs(~Goals + Urban.Rural, data=kids))
```

`littleSurvey` *Results from a little survey*

Description

These data are from a little survey given to a number of students in introductory statistics courses. Several of the items were prepared in multiple versions and distributed randomly to the students.

Format

A data frame with 279 observations on the following 20 variables.

- number a number between 1 and 30
- colorVer which version of the 'favorite color' question was on the survey. A factor with levels `v1` `v2`
- color favorite color if among predefined choices. A factor with levels `black` `green` `other` `purple` `red`
- otherColor favorite color if not among choices above.
- animalVer which version of the 'favorite color' question was on the survey. A factor with levels `v1` `v2`
- animal favorite animal if among predefined choices. A factor with levels `elephant` `giraffe` `lion` `other`.
- otherAnimal favorite animal if not among the predefined choices.
- pulseVer which version of the 'pulse' question was on the survey
- pulse self-reported pulse
- TVver which of three versions of the TV question was on the survey
- tvBox a factor with levels `<1` `>4` `>8` `1-2` `2-4` `4-8` `none` `other`
- tvHours a numeric vector

- surpriseVer which of two versions of the ’surprise’ question was on the survey
- surprise a factor with levels `no` `yes`
- playVer which of two versions of the ’play’ question was on the survey
- play a factor with levels `no` `yes`
- diseaseVer which of two versions of the ’play’ question was on the survey
- disease a factor with levels `A` `B`
- homeworkVer which of two versions of the ’homework’ question was on the survey
- homework a factor with levels `A` `B`

Examples

```
data(littleSurvey)
xtabs(~surprise+surpriseVer, data=littleSurvey)
xtabs(~disease+diseaseVer, data=littleSurvey)
```

`mathnoise` *Test performance and noise*

Description

In this experiment, hyperactive and control students were given a mathematics test in either a quiet or loud testing environment.

Format

A data frame with 40 observations on the following 3 variables.

- score score on a mathematics test
- noise a factor with levels `hi` `lo`
- group a factor with levels `control` `hyper`

Source

Sydney S. Zentall and Jandira H. Shaw, Effects of classroom noise on perfor- mance and activity of second-grade hyperactive and control children, *Journal of Educational Psychology* 72 (1980), no. 6, 830.

Examples

```
data(mathnoise)
xyplot(score~noise, data=mathnoise, group=group, type='a',
auto.key=list(columns=2, lines=TRUE, points=FALSE))
```

`miaa05` *MIAA basketball 2004-2005 season*

Description

Individual player statistics for the 2004-2005 Michigan Intercollegiate Athletic Association basketball season.

Format

A data frame with 134 observations on the following 27 variables.

- Number jersey number
- Player player's name
- GP games played
- GS games started
- Min minutes played
- AvgMin average minutes played per game
- FG field goals made
- FGA field goals attempted
- FGPct field goal percentage
- FG3 3-point field goals made
- FG3A 3-point field goals attempted
- FG3Pct 3-point field goal percentage
- FT free throws made
- FTA free throws attempted
- FTPct free throw percentage
- Off offensive rebounds
- Def defensive rebounds
- Tot total rebounds
- RBG rebounds per game
- PF personal fouls
- FO games fouled out
- A assists
- TO turn overs
- Blk blocked shots
- Stl steals
- Pts points scored
- PTSG points per game

Source

MIAA sports archives (http://www.miaa.org/)

Examples

```
data(miaa05)
histogram(~FTPct, data=miaa05)
```

---

mlb2004 *Major League Baseball 2004 team data*

Description

Team batting statistics, runs allowed, and runs scored for the 2004 Major League Baseball season.

Format

A data frame with 30 observations on the following 20 variables.

- Team team city, a factor
- League League, a factor with levels `AL` `NL`
- W number of wins
- L number of losses
- G number of games
- R number of runs scored
- OR oppnents' runs – number of runs allowed
- Rdiff run difference – `R - OR`
- AB number of at bats
- H number of hits
- DBL number of doubles
- TPL number of triples
- HR number of home runs
- BB number of walks (bases on balls)
- SO number of strike outs
- SB number of stolen bases
- CS number of times caught stealing
- BA batting average
- SLG slugging percentage
- OBA on base average

Examples

```
data(mlb2004)
xyplot(W ~ Rdiff, data=mlb2004)
```

---

ncaa2010 *NCAA Division I Basketball Results*

Description

Results of NCAA basketball games

Format

Seven variables describing NCAA Division I basketball games.

- date date on which game was played
- away visiting team
- ascore visiting team's score
- home home team
- hscore home team's score
- notes code indicting games played at neutral sites (n or N) or in tournaments (T)
- location where game was played

Source

kenpom.com

Examples

```
data(ncaa2010)
# add some additional variables to the data frame
ncaa2010$dscore <- ncaa2010$hscore- ncaa2010$ascore
ncaa2010$homeTeamWon <- ncaa2010$dscore > 0
ncaa2010$numHomeTeamWon <- -1 + 2 * as.numeric(ncaa2010$homeTeamWon)
w <- which(ncaa2010$homeTeamWon)
ncaa2010$winner <- as.character(ncaa2010$away)
ncaa2010$winner[w] <- as.character(ncaa2010$home)[w]
ncaa2010$loser <- as.character(ncaa2010$home)
ncaa2010$loser[w] <- as.character(ncaa2010$away)[w]
ncaa2010$homeTeamWon <- ncaa2010$winner == ncaa2010$home
ncaa2010$numHomeTeamWon <- -1 + 2 * as.numeric(ncaa2010$homeTeamWon)
```

nfl2007 *NFL 2007 season*

Description

Results of National Football League games (2007 season, including playoffs)

Format

A data frame with 267 observations on the following 7 variables.

- Date date on which game was played
- Visitor visiting team
- VisitorScore score for visiting team
- Home home team
- HomeScore score for home team
- Line 'betting line'
- TotalLine 'over/under' line (for combined score of both teams)

Examples

```
data(nfl2007); nfl <- nfl2007
nfl$dscore <- nfl$HomeScore - nfl$VisitorScore
w <- which(nfl$dscore > 0)
nfl$winner <- nfl$Visitor; nfl$winner[w] <- nfl$Home[w]
nfl$loser <- nfl$Home; nfl$loser[w] <- nfl$Visitor[w]
# did the home team win?
nfl$homeTeamWon <- nfl$dscore > 0
table(nfl$homeTeamWon)
table(nfl$dscore > nfl$line)
```

nlmax *Nonlinear maximization and minimization*

Description

`nlmin` and `nlmax` are thin wrappers around `nlm`, a non-linear minimizer. `nlmax` avoids the necessity of modifying the function to construct a minimization problem from a problem that is naturally a maximization problem. The `summary` method for the resulting objects provides output that is easier for humnans to read.

Usage

```
nlmax(f, ...)

nlmin(f, ...)

## S3 method for class 'nlmax'
summary(object, nsmall = 4, ...)

## S3 method for class 'nlmin'
summary(object, nsmall = 4, ...)
```

Arguments

`f`	a function to optimize
`object`	an object returned from `nlmin` or `nlmax`
`nsmall`	a numeric passed through to `format`
`...`	additional arguments passed to `nlm`. Note that `p` is a required argument for `nlm`. See the help for `nlm` for details.

Examples

```
summary( nlmax( function(x) 5 - 3*x - 5*x^2, p=0 ) )
```

`noise` *Noise – unfinished documentation*

Description

In order to test the effect of room noise, subjects were given a test under 5 diff sets of conditions: 1) no noise, 2) intermittent low volume, 3) intermittent high volume, 4) continuous low volume, and 5) continuous high volume.

Format

A data frame with 50 observations on the following 5 variables.

- id subject identifier
- score score on the test
- condition numeric code for condition
- volume a factor with levels `high` `low` `none`
- frequency a factor with levels `continuous` `intermittent` `none`

Examples

```
data(noise)
noise2 <- noise[noise$volume != 'none',]
model <- lm(score~volume*frequency, data=noise2)
anova(model)
xyplot(score~volume,noise2, groups=frequency, type='a',
auto.key=list(columns=2, points=FALSE, lines=TRUE))
```

`palettes` *Palette repair data*

Description

The palettes data set contains data from a firm that recycles palettes. Palettes from warehouses are bought, repaired, and resold. (Repairing a palette typically involves replacing one or two boards.) The company has four employees who do the repairs. The employer sampled five days for each employee and recorded the number of palettes repaired.

Format

A data frame with 20 observations on the following 3 variables.

- palettes number of palettes repaired
- employee a factor with levels `A` `B` `C` `D`
- day a factor with levels `day1` `day2` `day3` `day4` `day5`

Source

Michael Stob, Calvin College

Examples

```
data(palettes)
# Do the employees differ in the rate at which they repair palettes?
pal.lm1 <- lm(palettes~employee,palettes)
anova(pal.lm1)
# Now using day as a blocking variable
pal.lm2 <- lm(palettes~employee+day,palettes)
anova(pal.lm2)
xyplot(palettes~day, data=palettes,
groups=employee,
main="Productivity by day and employee",
type='b',auto.key=list(columns=4,points=FALSE,lines=TRUE))
```

paperplanes *Paper airplanes*

Description

Student-collected data from an experiment investigating the design of paper airplanes.

Format

A data frame with 16 observations on the following 5 variables.

- distance distance plane traveled (cm)
- paper type of paper used
- angle a numeric vector
- design design of plane (`hi performance` or `simple`)
- order order in which planes were thrown

Details

These data were collected by Stewart Fischer and David Tippetts, statistics students at the Queensland University of Technology in a subject taught by Dr. Margaret Mackisack. Here is their description of the data and its collection:

The experiment decided upon was to see if by using two different designs of paper aeroplane, how far the plane would travel. In considering this, the question arose, whether different types of paper and different angles of release would have any effect on the distance travelled. Knowing that paper aeroplanes are greatly influenced by wind, we had to find a way to eliminate this factor. We decided to perform the experiment in a hallway of the University, where the effects of wind can be controlled to some extent by closing doors.

In order to make the experimental units as homogeneous as possible we allocated one person to a task, so person 1 folded and threw all planes, person 2 calculated the random order assignment, measured all the distances, checked that the angles of flight were right, and checked that the plane release was the same each time.

The factors that we considered each had two levels as follows:

Paper: A4 size, 80g and 50g

Design: High Performance Dual Glider, and Incredibly Simple Glider (patterns attached to original report)

Angle of release: Horizontal, or 45 degrees upward.

The random order assignment was calculated using the random number function of a calculator. Each combination of factors was assigned a number from one to eight, the random numbers were generated and accordingly the order of the experiment was found.

Source

These data are also available at OzDASL, the Australasian Data and Story Library (http://www.statsci.org/data/).

References

Mackisack, M. S. (1994). What is the use of experiments conducted by statistics students? *Journal of Statistics Education*, 2, no 1.

Examples

```
data(paperplanes)
```

`pendulum` *Pendulum data*

Description

Period and pendulum length for a number of string and mass pendulums constructed by physics students. The same mass was used throughout, but the length of the string was varied from 10cm to 16 m.

Format

A data frame with 27 observations on the following 3 variables.

- length length of the pendulum (in meters)
- period average time of period (in seconds) over several swings of the pendulum
- delta.length an estimate of the accuracy of the length measurement

Source

Calvin College physics students under the direction of Professor Steve Plath.

Examples

```
data(pendulum)
xyplot(period ~ length, data=pendulum)
```

`petstress` *Pets and stress*

Description

Does having a pet or a friend cause more stress?

Format

A data frame with 45 observations on the following 2 variables.

- Group a factor with levels `Control`, `Friend`, or `Pet`
- Rate average heart rate while performing a stressful task

Details

Fourty-five women, all self-proclaimed dog-lovers, were randomly divided into three groups of subjects. Each performed a stressful task either alone, with a friend present, or with their dog present. The average heart rate during the task was used as a measure of stress.

Source

K. M. Allen, J. Blascovich, J. Tomaka, and R. M. Kelsey, Presence of human friends and pet dogs as moderators of autonomic responses to stress in women, *Journal of Personality and Social Psychology* 61 (1991), no. 4, 582–589.

References

These data also appear in

Brigitte Baldi and David S. Moore, *The Practice of Statistics in the Life Sciences*, Freeman, 2009.

Examples

```
data(petstress)
```

pheno *FUSION type 2 diabetes study*

Description

Phenotype and genotype data from the Finland United States Investigation of NIDDM (type 2) Diabetes (FUSION) study.

Format

Data frames with the following variables.

- id subject ID number for matching between data sets
- t2d a factor with levels `case` `control`
- bmi body mass index
- sex a factor with levels `F` `M`
- age age of subject at time phenotypes were colelcted
- smoker a factor with levels `former` `never` `occasional` `regular`
- chol total cholesterol
- waist waist circumference (cm)
- weight weight (kg)
- height height (cm)
- whr waist hip ratio
- sbp systolic blood pressure
- dbp diastolic blood pressure
- marker RS name of SNP
- markerID numeric ID for SNP
- allele1 first allele coded as 1=A, 2=C, 3=G, 4=T
- allele2 second allele coded as 1=A, 2=C, 3=G, 4=T
- genotype both alleles coded as a factor
- Adose number of A alleles
- Cdose number of C alleles
- Gdose number of G alleles
- Tdose number of T alleles

Source

Similar to the data presented in

Laura J. Scott, Karen L. Mohlke, Lori L. Bonnycastle, Cristen J. Willer, Yun Li, William L. Duren, Michael R. Erdos, Heather M. Stringham, Pe- ter S. Chines, Anne U. Jackson, Ludmila Prokunina-Olsson, Chia-Jen J. Ding, Amy J. Swift, Narisu Narisu, Tianle Hu, Randall Pruim, Rui Xiao, Xiao- Yi Y. Li, Karen N. Conneely, Nancy L. Riebow, Andrew G. Sprau, Maurine Tong, Peggy P. White, Kurt N. Hetrick, Michael W. Barnhart, Craig W. Bark, Janet L. Goldstein, Lee Watkins, Fang Xiang, Jouko Saramies, Thomas A. Buchanan, Richard M. Watanabe, Timo T. Valle, Leena Kinnunen, Goncalo R. Abecasis, Elizabeth W. Pugh, Kimberly F. Doheny, Richard N. Bergman, Jaakko Tuomilehto, Francis S. Collins, and Michael Boehnke, A genome-wide association study of type 2 diabetes in Finns detects multiple susceptibility vari- ants, *Science* (2007).

Examples

```
data(pheno); data(fusion1); data(fusion2)
fusion1m <- merge(fusion1, pheno, by="id", all.x=FALSE, all.y=FALSE)
xtabs(~t2d + genotype, data=fusion1m)
xtabs(~t2d + Gdose, data=fusion1m)
chisq.test( xtabs( ~t2d + genotype, data=fusion1m ) )
f1.glm <- glm( factor(t2d) ~ Gdose, data=fusion1m, family=binomial)
summary(f1.glm)
```

---

pigs *Pass the Pigs*

Description

This data set contains information collected from rolling the pair of pigs (found in the game "Pass the Pigs") 6000 times.

Format

A data frame with 6000 observations on the following 6 variables.

- roll roll number (1-6000)
- black numerical code for position of black pig
- blackF position of black pig coded as a factor
- pink numerical code for position of pink pig
- pinkF position of pink pig coded as a factor
- score score of the roll
- height height from which pigs were rolled (5 or 8 inches)
- start starting position of the pigs (0 = both pigs backwards, 1 = one bacwards one forwards, 2 = both forwards)

Details

In "Pass the Pigs", players roll two pig-shaped rubber dice and earn or lose points depending on the configuration of the rolled pigs. Players compete individually to earn 100 points. On each turn, a player rolls he or she decides to stop or until "pigging out" or

The pig configurations and their associated scores are

1 = Dot Up (0)

2 = Dot Down (0)

3 = Trotter (5)

4 = Razorback (5)

5 = Snouter (10)

6 = Leaning Jowler (15)

7 = Pigs are touching one another (-1; lose all points)

One pig Dot Up and one Dot Down ends the turn (a "pig out") and results in 0 points for the turn. If the pigs touch, the turn is ended and all points for the game must be forfeited. Two pigs in the Dot Up or Dot Down configuration score 1 point. Otherwise, The scores of the two pigs in different configurations are added together. The score is doubled if both both pigs have the same configuration, so, for example, two Snouters are worth 40 rather than 20.

The vector `pigConfig` is provided to assist in converting from the numeric codes above to pig configurations.

Source

John C. Kern II, Duquesne University (<`kern at mathcs.duq.edu`>)

Examples

```
data(pigs)
table(pigConfig[pigs$black])
```

`pitching2005` *Major League Baseball 2005 pitching*

Description

Major League Baseball pitching statistics for the 2005 season.

Format

A data frame with 653 observations on the following 27 variables.

- playerID unique identifier for each player
- yearID year
- stint for players who played with multiple teams in the same season, `stint` is increased by one each time the player joins a new team
- teamID three-letter identifier for team
- lgID league team plays in, coded as `AL` or `NL`
- W wins
- L losses
- G games played in
- GS games started
- CG complete games
- SHO shut outs
- SV saves recorded
- IPouts outs recorded (innings pitched, measured in outs rather than innings)
- H hits allowed
- ER earned runs allowed
- HR home runs allowed
- BB walks (bases on balls) allowed
- SO strike outs
- BAOpp opposing hitters' batting average
- ERA earned run average
- IBB intentional walks
- WP wild pitches
- HBP number of batters hit by pitch
- BK balks
- BFP batters faced pitching
- GF ratio of ground balls to fly balls
- R runs allowed

Examples

```
data(pitching2005)
xyplot(IPouts/3 ~ W, data=pitching2005, ylab="innings pitched", xlab="wins")
```

poison *Poison data*

Description

The data give the survival times (in hours) in a 3 x 4 factorial experiment, the factors being (a) three poisons and (b) four treatments. Each combination of the two factors is used for four animals. The allocation to animals is completely randomized.

Format

A data frame with 48 observations on the following 3 variables.

- Poison type of poison (1, 2, or 3)
- Treatment manner of treatment (1, 2, 3, or 4)
- Time time until death (hours)

Source

These data are also available from OzDASL, the Australian Data and Story Library (http://www.statsci.org/data/). (Note: The time measurements of the data at OzDASL are in units of tens of hours.)

References

Box, G. E. P., and Cox, D. R. (1964). An analysis of transformations (with Discussion). J. R. Statist. Soc. B, 26, 211-252.

Aitkin, M. (1987). Modelling variance heterogeneity in normal regression using GLIM. Appl. Statist., 36, 332-339.

Smyth, G. K., and Verbyla, A. P. (1999). Adjusted likelihood methods for modelling dispersion in generalized linear models. Environmetrics 10, 696-709. http://www.statsci.org/smyth/pubs/ties98tr.html.

Examples

```
data(poison)
poison.lm <- lm(Time~factor(Poison) * factor(Treatment), data=poison)
xplot(poison.lm,w=c(4,2))
anova(poison.lm)
# improved fit using a transformation
poison.lm2 <- lm(1/Time~factor(Poison) * factor(Treatment), data=poison)
xplot(poison.lm2,w=c(4,2))
anova(poison.lm)
```

`punting` *American football punting*

Description

Investigators studied physical characteristics and ability in 13 football punters. Each volunteer punted a football ten times. The investigators recorded the average distance for the ten punts, in feet. They also recorded the average hang time (time the ball is in the air before the receiver catches it), and a number of measures of leg strength and flexibility.

Format

A data frame with 13 observations on the following 7 variables.

- distance mean distance for 10 punts (feet)
- hang mean hang time (seconds)
- rStrength right leg strength (pounds)
- lStrength left leg strength (pounds)
- rFlexibility right leg flexibility (degrees)
- lFlexibility left leg flexibility (degrees)
- oStrength overall leg strength (foot-pounds)

Source

These data are also available at OzDASL (`http://www.statsci.org/data/`).

References

"The relationship between selected physical performance variables and football punting ability" by the Department of Health, Physical Education and Recreation at the Virginia Polytechnic Institute and State University, 1983.

Examples

```
data(punting)
xyplot(hang ~ distance, data=punting)
```

ratpoison *Rat poison – unfinished documentation*

Description

Data from an experiment to see whether flavor and location of rat poison influence the consumption by rats.

Format

A data frame with 20 observations on the following 3 variables.

- consumption a numeric vector
- flavor a factor with levels `bread` `butter-vanilla` `plain` `roast beef`
- location a factor with levels `A` `B` `C` `D` `E`

Examples

```
data(ratpoison)
```

rgolfballs *Simulated golf ball data*

Description

A matrix of random golf ball numbers simulated using `rmultinom(n=10000,size=486,prob=rep(0.25,4))`.

Examples

```
data(rgolfballs)
```

`rubberband` *Rubber band launching – unfinished documentation*

Description

Results of an experiment comparing a rubber band travels to the amount it was stretched prior to launch.

Format

A data frame with 16 observations on the following 2 variables.

- Stretch amount rubber band was stretched before launch
- Distance distance rubber band traveled

Examples

```
data(rubberband)
xyplot(Distance ~ Stretch, data=rubberband, type=c("p","r"))
```

`scent` *Maze tracing and scents*

Description

Subjects were asked to to complete a pencil and paper maze when they were smelling a floral scent and when they were not.

Format

A data frame with 21 observations on the following 12 variables.

- id ID number
- sex a factor with levels `F` and`M`
- smoker a factor with levels `N`, `Y`
- opinion opinion of the odor (`indiff`, `neg`, or `pos`)
- age age of subject (in years)
- first which treatment was first, `scented` or `unscented`
- u1 time (in seconds) in first unscented trial
- u2 time (in seconds) in second unscented trial
- u3 time (in seconds) in third unscented trial
- s1 time (in seconds) in first scented trial
- s2 time (in seconds) in second scented trial
- s3 time (in seconds) in third scented trial

Source

These data are also available at DASL, the data and story library (http://lib.stat.cmu.edu/DASL/).

References

Hirsch, A. R., and Johnston, L. H. "Odors and Learning," Smell \& Taste Treatment and Research Foundation, Chicago.

Examples

```
data(scent)
summary(scent)
```

---

snippet *Display or execute a snippet of R code*

Description

This command will display and/or execute small snippets of R code from the book *Foundations and Applications of Statistics: An Introduction Using R*.

Usage

```
snippet(name, execute = TRUE, view = !execute, echo = TRUE,
  ask = getOption("demo.ask"), verbose = getOption("verbose"),
  lib.loc = NULL, character.only = FALSE)
```

Arguments

name	name of snippet
execute	a logical. If TRUE, snippet code is executed. (The code and the results of the execution will be visible if echo is TRUE.)
view	a logical. If TRUE, snippet code is displayed 'as is'.
echo	a logical. If TRUE, show the R input when executing.
ask	a logical (or "default") indicating if devAskNewPage(ask=TRUE) should be called before graphical output happens from the snippet code. The value "default" (the factory-fresh default) means to ask if echo == TRUE and the graphics device appears to be interactive. This parameter applies both to any currently opened device and to any devices opened by the demo code. If this is evaluated to TRUE and the session is interactive, the user is asked to press RETURN to start.
verbose	a logical. If TRUE, additional diagnostics are printed.
lib.loc	character vector of directory names of R libraries, or NULL. The default value of NULL corresponds to all libraries currently known.
character.only	logical. If TRUE, use nameas character string.

Details

snippet works much like demo, but the interface is simplified.

Author(s)

Randall Pruim

See Also

demo, source.

---

soap *Dwindling soap*

Description

A bar of soap was weighed after showering to see how much soap was used each shower.

Format

A data frame with 15 observations on the following 3 variables.

- Date
- Day days since start of soap usage and data collection
- Weight weight of bar of soap (in grams)

Details

According to Rex Boggs:

I had a hypothesis that the daily weight of my bar of soap [in grams] in my shower wasn't a linear function, the reason being that the tiny little bar of soap at the end of its life seemed to hang around for just about ever. I wanted to throw it out, but I felt I shouldn't do so until it became unusable. And that seemed to take weeks.

Also I had recently bought some digital kitchen scales and felt I needed to use them to justify the cost. I hypothesized that the daily weight of a bar of soap might be dependent upon surface area, and hence would be a quadratic function . . . .

The data ends at day 22. On day 23 the soap broke into two pieces and one piece went down the plughole.

Source

Data collected by Rex Boggs and available from OzDASL (http://www.statsci.org/data/).

Examples

```
data(soap)
xyplot(Weight~Day, data=soap)
```

`spheres` *Measuring spheres*

Description

Measurements of the diameter (in meters) and mass (in kilograms) of a set of steel ball bearings.

Format

A data frame with 12 observations on the following 2 variables.

- diameter diameter of bearing (m)
- mass mass of the bearing (kg)

Source

These data were collected by Calvin College physics students under the direction of Steve Plath.

References

data(spheres)

`step` *Stepping experiment*

Description

An experiment was conducted by students at The Ohio State University in the fall of 1993 to explore the nature of the relationship between a person's heart rate and the frequency at which that person stepped up and down on steps of various heights.

Format

A data frame with 30 observations on the following 7 variables.

- order performance order
- block number of experimenter block
- restHR resting heart rate (beats per minute)
- HR final heart rate
- height height of step (`hi` or `lo`)
- freq whether subject stepped `fast`, `medium`, or `slow`

Details

An experiment was conducted by students at The Ohio State University in the fall of 1993 to explore the nature of the relationship between a person's heart rate and the frequency at which that person stepped up and down on steps of various heights. The response variable, heart rate, was measured in beats per minute. There were two different step heights: 5.75 inches (coded as `lo`), and 11.5 inches (coded as `hi`). There were three rates of stepping: 14 steps/min. (coded as `slow`), 21 steps/min. (coded as `medium`), and 28 steps/min. (coded as `fast`). This resulted in six possible height/frequency combinations. Each subject performed the activity for three minutes. Subjects were kept on pace by the beat of an electric metronome. One experimenter counted the subject's pulse for 20 seconds before and after each trial. The subject always rested between trials until her or his heart rate returned to close to the beginning rate. Another experimenter kept track of the time spent stepping. Each subject was always measured and timed by the same pair of experimenters to reduce variability in the experiment. Each pair of experimenters was treated as a block.

Source

These data are available at DASL, the data and story library (http://lib.stat.cmu.edu/DASL/).

Examples

```
data(step)
xyplot(HR-restHR ~ freq, data=step, groups=height, type='a')
xyplot(HR-restHR ~ height, data=step, groups=freq, type='a')
```

---

stereogram *Stereogram fusion*

Description

Results of an experiment on the effect of prior information on the time to fuse random dot stereograms. One group (NV) was given either no information or just verbal information about the shape of the embedded object. A second group (group VV) received both verbal information and visual information (e.g., a drawing of the object).

Format

A data frame with 78 observations on the following 2 variables.

- Time time until subject was able to fuse a random dot stereogram
- Group treatment group: `NV`(no visual instructions) `VV` (visual instructions)

Source

These data are available at DASL, the data and story library (http://lib.stat.cmu.edu/DASL/).

References

Frisby, J. P. and Clatworthy, J. L., "Learning to see complex random-dot stereograms," *Perception*, 4, (1975), pp. 173-178.

Cleveland, W. S. *Visualizing Data*. 1993.

Examples

```
data(stereogram)
require(Hmisc)
favstats(Time~Group, data=stereogram)
```

---

students — *Standardized test scores and GPAs*

Description

Standardized test scores and GPAs for 1000 students.

Format

A data frame with 1000 observations on the following 6 variables.

- ACT ACT score
- SAT SAT score
- Grad has the student graduated from college?
- GradGPA college GPA at graduation
- HSGPA high school GPA
- Cohort year of graduation or expected graduation

Examples

```
data(students)
xyplot(ACT ~ SAT, data=students)
xyplot(GradGPA ~ HSGPA, data=students)
```

tastetest *Taste test data*

Description

Tthe results from a study comparing different preparation methods for taste test samples.

Format

A data frame with 16 observations on 2 (`taste1`) or 4 (`tastetest`) variables.

- score taste score from a group of 50 testers
- scr a factor with levels `coarse` `fine`
- liq a factor with levels `hi` `lo`
- type a factor with levels `A` `B` `C` `D`

Details

The samples were prepared for tasting using either a coarse screen or a fine screen, and with either a high or low liquid content. A total taste score is recorded for each of 16 groups of 50 testers each. Each group had 25 men and 25 women, each of whom scored the samples on a scale from -3 (terrible) to 3 (excellent). The sum of these individual scores is the overall taste score for the group.

Source

E. Street and M. G. Carroll, *Preliminary evaluation of a food product*, Statistics: A Guide to the Unknown (Judith M. Tanur et al., eds.), Holden-Day, 1972, pp. 220-238.

Examples

```
data(tastetest)
data(taste1)
require(Hmisc)
xyplot(score~scr, data=tastetest)
xyplot(score~scr, groups=liq, tastetest, type='a')
favstats(score~scr, data=tastetest)
```

tdf — *Compute degrees of freedom for a 2-sample t-test*

Description

This function computes degrees of freedom for a 2-sample t-test from the standard deviations and sample sizes of the two samples.

Usage

```
tdf(sd1, sd2, n1, n2)
```

Arguments

sd1	standard deviation of the sample 1
sd2	standard deviation of the sample 2
n1	size of sample 1
n2	size of sample 2

Value

estimated degrees of freedom for 2-sample t-test

Examples

```
data(KidsFeet, package="mosaicData")
fs <- favstats( length ~ sex, data=KidsFeet ); fs
t.test( length ~ sex, data=KidsFeet )
tdf( fs[1,'sd'], fs[2,'sd'], fs[1,'n'], fs[2,'n'])
```

tirewear — *Estimating tirewear*

Description

Treadwear is estimated by two methods: weight loss and groove wear.

Format

A data frame with 16 observations on the following 2 variables.

- weight estimated wear (1000's of miles) base on weight loss
- groove estimated wear (1000's of miles) based on groove wear

Source

These data are available at DASL, the Data and Story Library (http://lib.stat.cmu.edu/DASL/).

References

R. D. Stichler, G. G. Richey, and J. Mandel, "Measurement of Treadware of Commercial Tires", *Rubber Age*, 73:2 (May 1953).

Examples

```
data(tirewear)
xyplot(weight ~ groove, data=tirewear)
```

`traffic` *New England traffic fatalities (1951-1959)*

Description

Used by Tufte as an example of the importance of context, these data show the traffic fatality rates in New England in the 1950s. Connecticut increased enforcement of speed limits in 1956. In their full context, it is difficult to say if the decline in Connecticut traffic fatalities from 1955 to 1956 can be attributed to the stricter enforcement.

Format

A data frame with 9 observations on the following 6 variables.

- year a year from 1951 to 1959
- cn.deaths number of traffic deaths in Connecticut
- ny deaths per 100,000 in New York
- cn deaths per 100,000 in Connecticut
- ma deaths per 100,000 in Massachusetts
- ri deaths per 100,000 in in Rhode Island

Source

Tufte, E. R. *The Visual Display of Quantitative Information*, 2nd ed. Graphics Press, 2001.

References

Donald T. Campbell and H. Laurence Ross. "The Connecticut Crackdown on Speeding: Time-Series Data in Quasi-Experimental Analysis", *Law \& Society Review* Vol. 3, No. 1 (Aug., 1968), pp. 33-54.

Gene V. Glass. "Analysis of Data on the Connecticut Speeding Crackdown as a Time-Series Quasi-Experiment" *Law & Society Review*, Vol. 3, No. 1 (Aug., 1968), pp. 55-76.

Examples

```
data(traffic)
xyplot(cn.deaths ~ year, data=traffic, type=c('l','g'))
trafficLong <- reshape(traffic,direction='long', idvar="year",
varying=list(3:6), v.names='deathRate',
times=names(traffic)[3:6], timevar='state')
xyplot(deathRate~year, groups=state, data=trafficLong, type='b',
auto.key=list(lines=TRUE, points=FALSE, columns=2))
```

`trebuchet` *Trebuchet data*

Description

Measurements from an experiment that involved firing projectiles with a small trebuchet under different conditions.

Format

Data frames with the following variables.

- object the object serving as projectilebean `big washerb` `bigWash` `BWB` `foose` `golf` `MWB` `SWB` `tennis ball` `wood`
- projectileWt weight of projectile (in grams)
- counterWt weight of counter weight (in kg)
- distance distance projectile traveled (in cm)
- form a factor with levels `a` `b` `B` `c` describing the configuration of the trebuchet.

Source

Data collected by Andrew Pruim as part of a Science Olympiad competition.

Examples

```
data(trebuchet); data(trebuchet1); data(trebuchet2)
xyplot(distance~projectileWt, data=trebuchet1)
xyplot(distance~projectileWt, data=trebuchet2)
xyplot(distance~projectileWt, groups=projectileWt, data=trebuchet)
```

`undocumented` *Undocumented functions*

Description

These objects are undocumented.

Details

In many cases, these are depricated in favor of alternative methods (often in the `mosaic` package) to obtain the same results but remain in the package to avoid breaking code used in Foundations and Applications of Statistics, the book associated with this package. For example, `funplot`, should be replaced by `plotFun` and `pdfplot`, `pmfhistogram`, and `pmfplot` should be replaced by `plotDist`.

In other cases, the functions are of limited suitability for general use.

Author(s)

Randall Pruim

`utilities` *Utilities bills*

Description

Data from utility bills at a residence.

Format

A data frame the following variables.

- month month (coded as a number)
- day day of month on which bill was calculated
- year year of bill
- temp average temperature (F) for billing period
- kwh electricity usage (kwh)
- ccf gas usage (ccf)
- thermsPerDay a numeric vector
- billingDays number of billing days in billing period
- totalbill total bill (in dollars)
- gasbill gas bill (in dollars)
- elecbill exectric bill (in dollars)
- notes notes about the billing period

- ccfpday average gas usage per day [`utilities2` only]
- kwhpday average electric usage per day [`utilities2` only]
- gasbillpday gas bill divided by billing days [`utilities2` only]
- elecbillpday electric bill divided by billing days a numeric vector [`utilities2` only]
- totalbillpday total bill divided by billing days a numeric vector [`utilities2` only]
- therms `thermsPerDay * billingDays` [`utilities2` only]
- monthsSinceY2K months since 2000 [`utilities2` only]

Source

Daniel T. Kaplan, *Statistical modeling: A fresh approach*, 2009.

Examples

```
data(utilities); data(utilities2)
xyplot(gasbill ~ temp, data=utilities)
xyplot(gasbillpday ~ temp, data=utilities2)
```

---

vaov *ANOVA vectors*

Description

Compute vectors associated with 1-way ANOVA

Usage

```
vaov(x, ...)

## S3 method for class 'formula'
vaov(x, data = parent.frame(), ...)
```

Arguments

`x`	a formula.
`data`	a data frame.
`...`	additional arguments.

Details

This is primarily designed for demonstration purposes to show how 1-way ANOVA models partition variance. It may not work properly for more complicated models.

Value

A data frame with variables including `grandMean`, `groupMean`, `ObsVsGrand`, `STotal`, `ObsVsGroup`, `SError`, `GroupVsGrand`, and `STreatment`. The usual SS terms can be computed from these by summing.

Author(s)

Randall Pruim

Examples

```
aov(pollution ~ location, data=airpollution)
vaov(pollution ~ location, data=airpollution)
```

`wilson.ci` *Confidence Intervals for Proportions*

Description

Alternatives to `prop.test` and `binom.test`.

Usage

```
wilson.ci(x, n = 100, conf.level = 0.95)
```

Arguments

`x`	number of 'successes'
`n`	number of trials
`conf.level`	confidence level

Details

`wald.ci` produces Wald confidence intervals. `wilson.ci` produces Wilson confidence intervals (also called "plus-4" confidence intervals) which are Wald intervals computed from data formed by adding 2 successes and 2 failures. The Wilson confidence intervals have better coverage rates for small samples.

Value

Lower and upper bounds of a two-sided confidence interval.

Author(s)

Randall Pruim

References

A. Agresti and B. A. Coull, Approximate is better then 'exact' for interval estimation of binomial proportions, *American Statistician* 52 (1998), 119–126.

Examples

```
prop.test(12,30)
prop.test(12,30, correct=FALSE)
wald.ci(12,30)
wilson.ci(12,30)
wald.ci(12+2,30+4)
```

---

workingWomen — *Women in the workforce*

Description

The labor force participation rate of women in each of 19 U.S. cities in each of two years. # Reference: United States Department of Labor Statistics # # Authorization: free use # # Description: # # Variable Names: # # 1. City: City in the United States # 2. labor72: Labor Force Participation rate of women in 1972 # 3. labor68: Labor Force Participation rate of women in 1968 # # The Data: #

Format

A data frame with 19 observations on the following 3 variables.

- city name of a U.S. city (coded as a factor with 19 levels)
- labor72 percent of women in labor force in 1972
- labor68 percent of women in labor force in 1968

Source

These data are from the United States Department of Labor Statistics and are also available at DASL, the Data and Story Library (http://lib.stat.cmu.edu/DASL/).

Examples

```
data(workingWomen)
xyplot(labor72 ~ labor68, workingWomen)
```

xplot *Augmented functions*

Description

These functions all behave similarly to the functions with the initial x removed from their names.

Usage

```
xplot(x, ...)

## Default S3 method:
xplot(...)

## S3 method for class 'lm'
xplot(x, which = c(1L:3, 5), caption = captions,
  panel.default = if (add.smooth) panel.xyplotsmooth else panel.xyplotpoints,
  sub.caption = NULL, main = "", print.plots = TRUE, ask = 1 <
  length(which) && dev.interactive(), type = "p",
  pch = trellis.par.get("plot.symbol")$pch,
  addline.col = trellis.par.get("add.line")$col,
  line.col = trellis.par.get("plot.line")$col,
  symbol.col = trellis.par.get("plot.symbol")$col,
  lty = trellis.par.get("superpose.line")$lty, ..., id.n = 3,
  labels.id = names(residuals(x)), cex.id = 0.7, qqline = TRUE,
  cook.levels = c(0.5, 1), add.smooth = TRUE, label.pos = c("left",
  "right"), cex.caption = 1)
```

Arguments

x	A numeric vector or formula.
which	A numeric vector indicating which plots to produce
caption	caption for the plot
panel.default	default panel function
sub.caption	secondary caption
main	as in xyplot
print.plots	a logical
ask	a logical
type	as in xyplot
pch	as in xyplot
addline.col	color for added lines
line.col	color for lines
symbol.col	color for symbols

`lty`	as in `xyplot`
`id.n`	a numeric
`labels.id`	a character vector of labels
`cex.id`	cex for ids
`qqline`	a logical
`cook.levels`	a logical
`add.smooth`	a logical
`label.pos`	position for labels, one of `"left"` or `"right"`
`cex.caption`	cex for the caption
`...`	arguments passed to other functions.
`panel`	a panel function

See Also

`plot`.

Examples

```
x <- runif(20)
xplot( lm ( 2*x + 5 + rnorm(20) ~ x ) )
```

Index

∗Topic **array**
col.perc, 10
∗Topic **datasets**
actgpa, 4
airlineArrival, 4
airpollution, 5
balldrop, 5
batting, 6
births78, 7
buckthorn, 8
bugs, 8
concrete, 10
corn, 11
cuckoo, 12
deathPenalty, 13
drag, 13
endurance, 14
familySmoking, 15
fumbles, 16
golfballs, 18
gpa, 19
heliumFootballs, 19
ice, 20
inflation, 22
Jordan8687, 22
kids, 23
littleSurvey, 24
mathnoise, 25
miaa05, 26
mlb2004, 27
ncaa2010, 28
nfl2007, 29
noise, 30
palettes, 31
paperplanes, 32
pendulum, 33
petstress, 34
pheno, 35
pigs, 36
pitching2005, 37
poison, 39
punting, 40
ratpoison, 41
rgolfballs, 41
rubberband, 42
scent, 42
soap, 44
spheres, 45
step, 45
stereogram, 46
students, 47
tastetest, 48
tirewear, 49
traffic, 50
trebuchet, 51
utilities, 52
workingWomen, 55
∗Topic **graphics**
col.fastR, 9
∗Topic **package**
fastR-package, 3
∗Topic **stats**
geolm, 17
vaov, 53
∗Topic **utilities**
snippet, 43

actgpa, 4
airlineArrival, 4
airpollution, 5
apply, *10*

balldrop, 5
batting, 6
births78, 7
buckthorn, 8
bugs, 8

col.fastR, 9

col.perc, 10
concrete, 10
concrete28 *(concrete)*, 10
concreteAll *(concrete)*, 10
corn, 11
cuckoo, 12

deathPen *(deathPenalty)*, 13
deathPenalty, 13
demo, *44*
drag, 13

endurance, 14

familySmoking, 15
fastR *(fastR-package)*, 3
fastR-package, 3
format, *30*
fumbles, 16
funplot *(undocumented)*, 52
funvec *(undocumented)*, 52
fusion1 *(pheno)*, 35
fusion2 *(pheno)*, 35

geolm, 17
givenOrder, 18
golfballs, 18
gpa, 19
grid.identify.points *(undocumented)*, 52

heliumFootballs, 19

ice, 20
inflation, 22

Jordan8687, 22

kids, 23

littleSurvey, 24
lm, *17, 18*

make.rm *(undocumented)*, 52
mathnoise, 25
miaa05, 26
mlb2004, 27
multirandom *(undocumented)*, 52

ncaa2008 *(ncaa2010)*, 28
ncaa2009 *(ncaa2010)*, 28
ncaa2010, 28
nfl2007, 29
nlm, *29, 30*
nlmax, 29
nlmax, *(nlmax)*, 29
nlmin *(nlmax)*, 29
noise, 30
ntigram *(undocumented)*, 52

palettes, 31
panel.lm *(undocumented)*, 52
panel.smooth *(undocumented)*, 52
panel.xyplotpoints *(undocumented)*, 52
panel.xyplotsmooth *(undocumented)*, 52
paperplanes, 32
pdfplot *(undocumented)*, 52
pendulum, 33
petstress, 34
pheno, 35
pigConfig *(pigs)*, 36
pigs, 36
pitching2005, 37
plot, *57*
pmfhistogram *(undocumented)*, 52
pmfplot *(undocumented)*, 52
poison, 39
print.stemplot *(undocumented)*, 52
probplot *(undocumented)*, 52
punting, 40

ratpoison, 41
rgolfballs, 41
row.perc *(col.perc)*, 10
rubberband, 42

scent, 42
show.fumbles *(undocumented)*, 52
show.poisson *(undocumented)*, 52
show.settings, *9*
snippet, 43
soap, 44
source, *44*
spheres, 45
stemplot *(undocumented)*, 52
step, 45
stereogram, 46
students, 47
summary.nlmax *(nlmax)*, 29
summary.nlmin *(nlmax)*, 29

taste1 *(tastetest)*, 48

tastetest, 48
tdf, 49
tirewear, 49
to2d *(*geolm*)*, 17
traffic, 50
trebuchet, 51
trebuchet1 *(*trebuchet*)*, 51
trebuchet2 *(*trebuchet*)*, 51
trellis.par.set, *9*

undocumented, 52
utilities, 52
utilities2 *(*utilities*)*, 52

vaov, 53

wald.ci *(*wilson.ci*)*, 54
wilson.ci, 54
workingWomen, 55

xplot, 56
xtabs, *10*