Package 'MASS'

April 26, 2024

Priority recommended

Version 7.3-60.2

Date 2024-01-12

Revision $Rev: 3641 $

Depends R (>= 4.4.0), grDevices, graphics, stats, utils

Imports methods

Suggests lattice, nlme, nnet, survival

Description Functions and datasets to support Venables and Ripley,
``Modern Applied Statistics with S'' (4th edition, 2002).

Title Support Functions and Datasets for Venables and Ripley's MASS

LazyData yes

ByteCompile yes

License GPL-2 | GPL-3

URL http://www.stats.ox.ac.uk/pub/MASS4/

Contact <MASS@stats.ox.ac.uk>

NeedsCompilation yes

Author Brian Ripley [aut, cre, cph],
Bill Venables [ctb],
Douglas M. Bates [ctb],
Kurt Hornik [trl] (partial port ca 1998),
Albrecht Gebhardt [trl] (partial port ca 1998),
David Firth [ctb]

Maintainer Brian Ripley <ripley@stats.ox.ac.uk>

Repository CRAN

Date/Publication 2024-04-26 12:02:47 UTC

R topics documented:

`abbey` *Determinations of Nickel Content*

Description

A numeric vector of 31 determinations of nickel content (ppm) in a Canadian syenite rock.

Usage

```
abbey
```

Source

S. Abbey (1988) *Geostandards Newsletter* **12**, 241.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

`accdeaths` *Accidental Deaths in the US 1973-1978*

Description

A regular time series giving the monthly totals of accidental deaths in the USA.

Usage

```
accdeaths
```

Details

The values for first six months of 1979 (p. 326) were `7798` `7406` `8363` `8460` `9217` `9316`.

Source

P. J. Brockwell and R. A. Davis (1991) *Time Series: Theory and Methods.* Springer, New York.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

`addterm` *Try All One-Term Additions to a Model*

Description

Try fitting all models that differ from the current model by adding a single term from those supplied, maintaining marginality.

This function is generic; there exist methods for classes `lm` and `glm` and the default method will work for many other classes.

Usage

```
addterm(object, ...)

## Default S3 method:
addterm(object, scope, scale = 0, test = c("none", "Chisq"),
        k = 2, sorted = FALSE, trace = FALSE, ...)
## S3 method for class 'lm'
addterm(object, scope, scale = 0, test = c("none", "Chisq", "F"),
        k = 2, sorted = FALSE, ...)
## S3 method for class 'glm'
addterm(object, scope, scale = 0, test = c("none", "Chisq", "F"),
        k = 2, sorted = FALSE, trace = FALSE, ...)
```

Arguments

`object`	An object fitted by some model-fitting function.
`scope`	a formula specifying a maximal model which should include the current one. All additional terms in the maximal model with all marginal terms in the original model are tried.
`scale`	used in the definition of the AIC statistic for selecting the models, currently only for `lm`, `aov` and `glm` models. Specifying `scale` asserts that the residual standard error or dispersion is known.
`test`	should the results include a test statistic relative to the original model? The F test is only appropriate for `lm` and `aov` models, and perhaps for some over-dispersed `glm` models. The Chisq test can be an exact test (`lm` models with known scale) or a likelihood-ratio test depending on the method.

k	the multiple of the number of degrees of freedom used for the penalty. Only k=2 gives the genuine AIC: k = log(n) is sometimes referred to as BIC or SBC.
sorted	should the results be sorted on the value of AIC?
trace	if TRUE additional information may be given on the fits as they are tried.
...	arguments passed to or from other methods.

Details

The definition of AIC is only up to an additive constant: when appropriate (lm models with specified scale) the constant is taken to be that used in Mallows' Cp statistic and the results are labelled accordingly.

Value

A table of class "anova" containing at least columns for the change in degrees of freedom and AIC (or Cp) for the models. Some methods will give further information, for example sums of squares, deviances, log-likelihoods and test statistics.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

dropterm, stepAIC

Examples

```
quine.hi <- aov(log(Days + 2.5) ~ .^4, quine)
quine.lo <- aov(log(Days+2.5) ~ 1, quine)
addterm(quine.lo, quine.hi, test="F")

house.glm0 <- glm(Freq ~ Infl*Type*Cont + Sat, family=poisson,
                  data=housing)
addterm(house.glm0, ~. + Sat:(Infl+Type+Cont), test="Chisq")
house.glm1 <- update(house.glm0, . ~ . + Sat*(Infl+Type+Cont))
addterm(house.glm1, ~. + Sat:(Infl+Type+Cont)^2, test = "Chisq")
```

Aids2 *Australian AIDS Survival Data*

Description

Data on patients diagnosed with AIDS in Australia before 1 July 1991.

Usage

```
Aids2
```

Format

This data frame contains 2843 rows and the following columns:

`state` Grouped state of origin: `"NSW"` includes ACT and `"other"` is WA, SA, NT and TAS.

`sex` Sex of patient.

`diag` (Julian) date of diagnosis.

`death` (Julian) date of death or end of observation.

`status` `"A"` (alive) or `"D"` (dead) at end of observation.

`T.categ` Reported transmission category.

`age` Age (years) at diagnosis.

Note

This data set has been slightly jittered as a condition of its release, to ensure patient confidentiality.

Source

Dr P. J. Solomon and the Australian National Centre in HIV Epidemiology and Clinical Research.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

`Animals` *Brain and Body Weights for 28 Species*

Description

Average brain and body weights for 28 species of land animals.

Usage

```
Animals
```

Format

`body` body weight in kg.

`brain` brain weight in g.

Note

The name `Animals` avoided conflicts with a system dataset `animals` in S-PLUS 4.5 and later.

Source

P. J. Rousseeuw and A. M. Leroy (1987) *Robust Regression and Outlier Detection.* Wiley, p. 57.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

`anorexia` *Anorexia Data on Weight Change*

Description

The `anorexia` data frame has 72 rows and 3 columns. Weight change data for young female anorexia patients.

Usage

```
anorexia
```

Format

This data frame contains the following columns:

`Treat` Factor of three levels: `"Cont"` (control), `"CBT"` (Cognitive Behavioural treatment) and `"FT"` (family treatment).

`Prewt` Weight of patient before study period, in lbs.

`Postwt` Weight of patient after study period, in lbs.

Source

Hand, D. J., Daly, F., McConway, K., Lunn, D. and Ostrowski, E. eds (1993) *A Handbook of Small Data Sets.* Chapman & Hall, Data set 285 (p. 229)

(Note that the original source mistakenly says that weights are in kg.)

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

anova.negbin *Likelihood Ratio Tests for Negative Binomial GLMs*

Description

Method function to perform sequential likelihood ratio tests for Negative Binomial generalized linear models.

Usage

```
## S3 method for class 'negbin'
anova(object, ..., test = "Chisq")
```

Arguments

object	Fitted model object of class "negbin", inheriting from classes "glm" and "lm", specifying a Negative Binomial fitted GLM. Typically the output of glm.nb().
...	Zero or more additional fitted model objects of class "negbin". They should form a nested sequence of models, but need not be specified in any particular order.
test	Argument to match the test argument of anova.glm. Ignored (with a warning if changed) if a sequence of two or more Negative Binomial fitted model objects is specified, but possibly used if only one object is specified.

Details

This function is a method for the generic function anova() for class "negbin". It can be invoked by calling anova(x) for an object x of the appropriate class, or directly by calling anova.negbin(x) regardless of the class of the object.

Note

If only one fitted model object is specified, a sequential analysis of deviance table is given for the fitted model. The theta parameter is kept fixed. If more than one fitted model object is specified they must all be of class "negbin" and likelihood ratio tests are done of each model within the next. In this case theta is assumed to have been re-estimated for each model.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

glm.nb, negative.binomial, summary.negbin

Examples

```
m1 <- glm.nb(Days ~ Eth*Age*Lrn*Sex, quine, link = log)
m2 <- update(m1, . ~ . - Eth:Age:Lrn:Sex)
anova(m2, m1)
anova(m2)
```

---

area *Adaptive Numerical Integration*

Description

Integrate a function of one variable over a finite range using a recursive adaptive method. This function is mainly for demonstration purposes.

Usage

```
area(f, a, b, ..., fa = f(a, ...), fb = f(b, ...),
     limit = 10, eps = 1e-05)
```

Arguments

f	The integrand as an S function object. The variable of integration must be the first argument.
a	Lower limit of integration.
b	Upper limit of integration.
...	Additional arguments needed by the integrand.
fa	Function value at the lower limit.
fb	Function value at the upper limit.
limit	Limit on the depth to which recursion is allowed to go.
eps	Error tolerance to control the process.

Details

The method divides the interval in two and compares the values given by Simpson's rule and the trapezium rule. If these are within eps of each other the Simpson's rule result is given, otherwise the process is applied separately to each half of the interval and the results added together.

Value

The integral from a to b of f(x).

References

Venables, W. N. and Ripley, B. D. (1994) *Modern Applied Statistics with S-Plus.* Springer. pp. 105–110.

Examples

```
area(sin, 0, pi)  # integrate the sin function from 0 to pi.
```

`bacteria` *Presence of Bacteria after Drug Treatments*

Description

Tests of the presence of the bacteria *H. influenzae* in children with otitis media in the Northern Territory of Australia.

Usage

```
bacteria
```

Format

This data frame has 220 rows and the following columns:

y presence or absence: a factor with levels `n` and `y`.

ap active/placebo: a factor with levels `a` and `p`.

hilo hi/low compliance: a factor with levels `hi` amd `lo`.

week numeric: week of test.

ID subject ID: a factor.

trt a factor with levels `placebo`, `drug` and `drug+`, a re-coding of `ap` and `hilo`.

Details

Dr A. Leach tested the effects of a drug on 50 children with a history of otitis media in the Northern Territory of Australia. The children were randomized to the drug or the a placebo, and also to receive active encouragement to comply with taking the drug.

The presence of *H. influenzae* was checked at weeks 0, 2, 4, 6 and 11: 30 of the checks were missing and are not included in this data frame.

Source

Dr Amanda Leach *via* Mr James McBroom.

References

Menzies School of Health Research 1999–2000 Annual Report. p.20. https://www.menzies.edu.au/icms_docs/172302_2000_Annual_report.pdf.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

Examples

```
contrasts(bacteria$trt) <- structure(contr.sdif(3),
     dimnames = list(NULL, c("drug", "encourage")))
## fixed effects analyses
## IGNORE_RDIFF_BEGIN
summary(glm(y ~ trt * week, binomial, data = bacteria))
summary(glm(y ~ trt + week, binomial, data = bacteria))
summary(glm(y ~ trt + I(week > 2), binomial, data = bacteria))
## IGNORE_RDIFF_END

# conditional random-effects analysis
library(survival)
bacteria$Time <- rep(1, nrow(bacteria))
coxph(Surv(Time, unclass(y)) ~ week + strata(ID),
      data = bacteria, method = "exact")
coxph(Surv(Time, unclass(y)) ~ factor(week) + strata(ID),
      data = bacteria, method = "exact")
coxph(Surv(Time, unclass(y)) ~ I(week > 2) + strata(ID),
      data = bacteria, method = "exact")

# PQL glmm analysis
library(nlme)
## IGNORE_RDIFF_BEGIN
summary(glmmPQL(y ~ trt + I(week > 2), random = ~ 1 | ID,
                family = binomial, data = bacteria))
## IGNORE_RDIFF_END
```

`bandwidth.nrd` *Bandwidth for density() via Normal Reference Distribution*

Description

A well-supported rule-of-thumb for choosing the bandwidth of a Gaussian kernel density estimator.

Usage

```
bandwidth.nrd(x)
```

Arguments

`x`	A data vector.

Value

A bandwidth on a scale suitable for the `width` argument of `density`.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Springer, equation (5.5) on page 130.

Examples

```
# The function is currently defined as
function(x)
{
    r <- quantile(x, c(0.25, 0.75))
    h <- (r[2] - r[1])/1.34
    4 * 1.06 * min(sqrt(var(x)), h) * length(x)^(-1/5)
}
```

---

bcv *Biased Cross-Validation for Bandwidth Selection*

Description

Uses biased cross-validation to select the bandwidth of a Gaussian kernel density estimator.

Usage

```
bcv(x, nb = 1000, lower, upper)
```

Arguments

x	a numeric vector
nb	number of bins to use.
lower, upper	Range over which to minimize. The default is almost always satisfactory.

Value

a bandwidth

References

Scott, D. W. (1992) *Multivariate Density Estimation: Theory, Practice, and Visualization.* Wiley.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

See Also

ucv, width.SJ, density

Examples

```
bcv(geyser$duration)
```

beav1 *Body Temperature Series of Beaver 1*

Description

Reynolds (1994) describes a small part of a study of the long-term temperature dynamics of beaver *Castor canadensis* in north-central Wisconsin. Body temperature was measured by telemetry every 10 minutes for four females, but data from a one period of less than a day for each of two animals is used there.

Usage

```
beav1
```

Format

The beav1 data frame has 114 rows and 4 columns. This data frame contains the following columns:

`day` Day of observation (in days since the beginning of 1990), December 12–13.

`time` Time of observation, in the form `0330` for 3.30am.

`temp` Measured body temperature in degrees Celsius.

`activ` Indicator of activity outside the retreat.

Note

The observation at 22:20 is missing.

Source

P. S. Reynolds (1994) Time-series analyses of beaver body temperatures. Chapter 11 of Lange, N., Ryan, L., Billard, L., Brillinger, D., Conquest, L. and Greenhouse, J. eds (1994) *Case Studies in Biometry.* New York: John Wiley and Sons.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

See Also

beav2

Examples

```
beav1 <- within(beav1,
               hours <- 24*(day-346) + trunc(time/100) + (time%%100)/60)
plot(beav1$hours, beav1$temp, type="l", xlab="time",
   ylab="temperature", main="Beaver 1")
usr <- par("usr"); usr[3:4] <- c(-0.2, 8); par(usr=usr)
lines(beav1$hours, beav1$activ, type="s", lty=2)
temp <- ts(c(beav1$temp[1:82], NA, beav1$temp[83:114]),
           start = 9.5, frequency = 6)
activ <- ts(c(beav1$activ[1:82], NA, beav1$activ[83:114]),
            start = 9.5, frequency = 6)

acf(temp[1:53])
acf(temp[1:53], type = "partial")
ar(temp[1:53])
act <- c(rep(0, 10), activ)
X <- cbind(1, act = act[11:125], act1 = act[10:124],
          act2 = act[9:123], act3 = act[8:122])
alpha <- 0.80
stemp <- as.vector(temp - alpha*lag(temp, -1))
sX <- X[-1, ] - alpha * X[-115,]
beav1.ls <- lm(stemp ~ -1 + sX, na.action = na.omit)
summary(beav1.ls, correlation = FALSE)
rm(temp, activ)
```

---

beav2 *Body Temperature Series of Beaver 2*

Description

Reynolds (1994) describes a small part of a study of the long-term temperature dynamics of beaver *Castor canadensis* in north-central Wisconsin. Body temperature was measured by telemetry every 10 minutes for four females, but data from a one period of less than a day for each of two animals is used there.

Usage

```
beav2
```

Format

The `beav2` data frame has 100 rows and 4 columns. This data frame contains the following columns:

`day` Day of observation (in days since the beginning of 1990), November 3–4.

`time` Time of observation, in the form `0330` for 3.30am.

`temp` Measured body temperature in degrees Celsius.

`activ` Indicator of activity outside the retreat.

Source

P. S. Reynolds (1994) Time-series analyses of beaver body temperatures. Chapter 11 of Lange, N., Ryan, L., Billard, L., Brillinger, D., Conquest, L. and Greenhouse, J. eds (1994) *Case Studies in Biometry.* New York: John Wiley and Sons.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

See Also

beav1

Examples

```
attach(beav2)
beav2$hours <- 24*(day-307) + trunc(time/100) + (time%%100)/60
plot(beav2$hours, beav2$temp, type = "l", xlab = "time",
   ylab = "temperature", main = "Beaver 2")
usr <- par("usr"); usr[3:4] <- c(-0.2, 8); par(usr = usr)
lines(beav2$hours, beav2$activ, type = "s", lty = 2)

temp <- ts(temp, start = 8+2/3, frequency = 6)
activ <- ts(activ, start = 8+2/3, frequency = 6)
acf(temp[activ == 0]); acf(temp[activ == 1]) # also look at PACFs
ar(temp[activ == 0]); ar(temp[activ == 1])

arima(temp, order = c(1,0,0), xreg = activ)
dreg <- cbind(sin = sin(2*pi*beav2$hours/24), cos = cos(2*pi*beav2$hours/24))
arima(temp, order = c(1,0,0), xreg = cbind(active=activ, dreg))

## IGNORE_RDIFF_BEGIN
library(nlme) # for gls and corAR1
beav2.gls <- gls(temp ~ activ, data = beav2, correlation = corAR1(0.8),
                 method = "ML")
summary(beav2.gls)
summary(update(beav2.gls, subset = 6:100))
detach("beav2"); rm(temp, activ)
## IGNORE_RDIFF_END
```

Belgian-phones *Belgium Phone Calls 1950-1973*

Description

A list object with the annual numbers of telephone calls, in Belgium. The components are:

`year` last two digits of the year.

`calls` number of telephone calls made (in millions of calls).

Usage

```
phones
```

Source

P. J. Rousseeuw and A. M. Leroy (1987) *Robust Regression & Outlier Detection.* Wiley.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

---

biopsy *Biopsy Data on Breast Cancer Patients*

Description

This breast cancer database was obtained from the University of Wisconsin Hospitals, Madison from Dr. William H. Wolberg. He assessed biopsies of breast tumours for 699 patients up to 15 July 1992; each of nine attributes has been scored on a scale of 1 to 10, and the outcome is also known. There are 699 rows and 11 columns.

Usage

```
biopsy
```

Format

This data frame contains the following columns:

`ID` sample code number (not unique).

`V1` clump thickness.

`V2` uniformity of cell size.

`V3` uniformity of cell shape.

`V4` marginal adhesion.

`V5` single epithelial cell size.

`V6` bare nuclei (16 values are missing).

`V7` bland chromatin.

`V8` normal nucleoli.

`V9` mitoses.

`class` `"benign"` or `"malignant"`.

Source

P. M. Murphy and D. W. Aha (1992). UCI Repository of machine learning databases. [Machine-readable data repository]. Irvine, CA: University of California, Department of Information and Computer Science.

O. L. Mangasarian and W. H. Wolberg (1990) Cancer diagnosis via linear programming. *SIAM News* **23**, pp 1 & 18.

William H. Wolberg and O.L. Mangasarian (1990) Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proceedings of the National Academy of Sciences, U.S.A.* **87**, pp. 9193–9196.

O. L. Mangasarian, R. Setiono and W.H. Wolberg (1990) Pattern recognition via linear programming: Theory and application to medical diagnosis. In *Large-scale Numerical Optimization* eds Thomas F. Coleman and Yuying Li, SIAM Publications, Philadelphia, pp 22–30.

K. P. Bennett and O. L. Mangasarian (1992) Robust linear programming discrimination of two linearly inseparable sets. *Optimization Methods and Software* **1**, pp. 23–34 (Gordon & Breach Science Publishers).

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

`birthwt` *Risk Factors Associated with Low Infant Birth Weight*

Description

The `birthwt` data frame has 189 rows and 10 columns. The data were collected at Baystate Medical Center, Springfield, Mass during 1986.

Usage

```
birthwt
```

Format

This data frame contains the following columns:

`low` indicator of birth weight less than 2.5 kg.

`age` mother's age in years.

`lwt` mother's weight in pounds at last menstrual period.

`race` mother's race (`1` = white, `2` = black, `3` = other).

`smoke` smoking status during pregnancy.

`ptl` number of previous premature labours.

`ht` history of hypertension.

`ui` presence of uterine irritability.

`ftv` number of physician visits during the first trimester.

`bwt` birth weight in grams.

Source

Hosmer, D.W. and Lemeshow, S. (1989) *Applied Logistic Regression.* New York: Wiley

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

Examples

```
bwt <- with(birthwt, {
race <- factor(race, labels = c("white", "black", "other"))
ptd <- factor(ptl > 0)
ftv <- factor(ftv)
levels(ftv)[-(1:2)] <- "2+"
data.frame(low = factor(low), age, lwt, race, smoke = (smoke > 0),
           ptd, ht = (ht > 0), ui = (ui > 0), ftv)
})
options(contrasts = c("contr.treatment", "contr.poly"))
glm(low ~ ., binomial, bwt)
```

Boston *Housing Values in Suburbs of Boston*

Description

The `Boston` data frame has 506 rows and 14 columns.

Usage

```
Boston
```

Format

This data frame contains the following columns:

`crim` per capita crime rate by town.

`zn` proportion of residential land zoned for lots over 25,000 sq.ft.

`indus` proportion of non-retail business acres per town.

`chas` Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).

`nox` nitrogen oxides concentration (parts per 10 million).

`rm` average number of rooms per dwelling.

`age` proportion of owner-occupied units built prior to 1940.

`dis` weighted mean of distances to five Boston employment centres.

`rad` index of accessibility to radial highways.

`tax` full-value property-tax rate per $10,000.

`ptratio` pupil-teacher ratio by town.

`black` $1000(Bk - 0.63)^2$ where Bk is the proportion of blacks by town.

`lstat` lower status of the population (percent).

`medv` median value of owner-occupied homes in $1000s.

Source

Harrison, D. and Rubinfeld, D.L. (1978) Hedonic prices and the demand for clean air. *J. Environ. Economics and Management* **5**, 81–102.

Belsley D.A., Kuh, E. and Welsch, R.E. (1980) *Regression Diagnostics. Identifying Influential Data and Sources of Collinearity.* New York: Wiley.

`boxcox` *Box-Cox Transformations for Linear Models*

Description

Computes and optionally plots profile log-likelihoods for the parameter of the Box-Cox power transformation.

Usage

```
boxcox(object, ...)

## Default S3 method:
boxcox(object, lambda = seq(-2, 2, 1/10), plotit = TRUE,
       interp, eps = 1/50, xlab = expression(lambda),
       ylab = "log-Likelihood", ...)

## S3 method for class 'formula'
boxcox(object, lambda = seq(-2, 2, 1/10), plotit = TRUE,
       interp, eps = 1/50, xlab = expression(lambda),
       ylab = "log-Likelihood", ...)

## S3 method for class 'lm'
boxcox(object, lambda = seq(-2, 2, 1/10), plotit = TRUE,
       interp, eps = 1/50, xlab = expression(lambda),
       ylab = "log-Likelihood", ...)
```

Arguments

object	a formula or fitted model object. Currently only lm and aov objects are handled.
lambda	vector of values of lambda – default $(-2, 2)$ in steps of 0.1.
plotit	logical which controls whether the result should be plotted.
interp	logical which controls whether spline interpolation is used. Default to TRUE if plotting with lambda of length less than 100.
eps	Tolerance for lambda = 0; defaults to 0.02.
xlab	defaults to "lambda".
ylab	defaults to "log-Likelihood".
...	additional parameters to be used in the model fitting.

Value

A list of the lambda vector and the computed profile log-likelihood vector, invisibly if the result is plotted.

Side Effects

If plotit = TRUE plots log-likelihood *vs* lambda and indicates a 95% confidence interval about the maximum observed value of lambda. If interp = TRUE, spline interpolation is used to give a smoother plot.

References

Box, G. E. P. and Cox, D. R. (1964) An analysis of transformations (with discussion). *Journal of the Royal Statistical Society B*, **26**, 211–252.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

Examples

```
boxcox(Volume ~ log(Height) + log(Girth), data = trees,
       lambda = seq(-0.25, 0.25, length.out = 10))

boxcox(Days+1 ~ Eth*Sex*Age*Lrn, data = quine,
       lambda = seq(-0.05, 0.45, length.out = 20))
```

---

cabbages *Data from a cabbage field trial*

Description

The cabbages data set has 60 observations and 4 variables

Usage

```
cabbages
```

Format

This data frame contains the following columns:

`Cult` Factor giving the cultivar of the cabbage, two levels: `c39` and `c52`.

`Date` Factor specifying one of three planting dates: `d16`, `d20` or `d21`.

`HeadWt` Weight of the cabbage head, presumably in kg.

`VitC` Ascorbic acid content, in undefined units.

Source

Rawlings, J. O. (1988) *Applied Regression Analysis: A Research Tool.* Wadsworth and Brooks/Cole. Example 8.4, page 219. (Rawlings cites the original source as the files of the late Dr Gertrude M Cox.)

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

---

caith *Colours of Eyes and Hair of People in Caithness*

Description

Data on the cross-classification of people in Caithness, Scotland, by eye and hair colour. The region of the UK is particularly interesting as there is a mixture of people of Nordic, Celtic and Anglo-Saxon origin.

Usage

```
caith
```

Format

A 4 by 5 table with rows the eye colours (blue, light, medium, dark) and columns the hair colours (fair, red, medium, dark, black).

Source

Fisher, R.A. (1940) The precision of discriminant functions. *Annals of Eugenics (London)* **10**, 422–429.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

Examples

```
## IGNORE_RDIFF_BEGIN
## The signs can vary by platform
corresp(caith)
## IGNORE_RDIFF_END
dimnames(caith)[[2]] <- c("F", "R", "M", "D", "B")
par(mfcol=c(1,3))
plot(corresp(caith, nf=2)); title("symmetric")
plot(corresp(caith, nf=2), type="rows"); title("rows")
plot(corresp(caith, nf=2), type="col"); title("columns")
par(mfrow=c(1,1))
```

`Cars93` *Data from 93 Cars on Sale in the USA in 1993*

Description

The `Cars93` data frame has 93 rows and 27 columns.

Usage

```
Cars93
```

Format

This data frame contains the following columns:

`Manufacturer` Manufacturer.

`Model` Model.

`Type` Type: a factor with levels `"Small"`, `"Sporty"`, `"Compact"`, `"Midsize"`, `"Large"` and `"Van"`.

`Min.Price` Minimum Price (in $1,000): price for a basic version.

`Price` Midrange Price (in $1,000): average of `Min.Price` and `Max.Price`.

`Max.Price` Maximum Price (in $1,000): price for "a premium version".

`MPG.city` City MPG (miles per US gallon by EPA rating).

`MPG.highway` Highway MPG.

`AirBags` Air Bags standard. Factor: none, driver only, or driver & passenger.

`DriveTrain` Drive train type: rear wheel, front wheel or 4WD; (factor).

`Cylinders` Number of cylinders (missing for Mazda RX-7, which has a rotary engine).

`EngineSize` Engine size (litres).

`Horsepower` Horsepower (maximum).

`RPM` RPM (revs per minute at maximum horsepower).

`Rev.per.mile` Engine revolutions per mile (in highest gear).

`Man.trans.avail` Is a manual transmission version available? (yes or no, Factor).

`Fuel.tank.capacity` Fuel tank capacity (US gallons).

`Passengers` Passenger capacity (persons)

`Length` Length (inches).

`Wheelbase` Wheelbase (inches).

`Width` Width (inches).

`Turn.circle` U-turn space (feet).

`Rear.seat.room` Rear seat room (inches) (missing for 2-seater vehicles).

`Luggage.room` Luggage capacity (cubic feet) (missing for vans).

`Weight` Weight (pounds).

`Origin` Of non-USA or USA company origins? (factor).

`Make` Combination of Manufacturer and Model (character).

Details

Cars were selected at random from among 1993 passenger car models that were listed in both the *Consumer Reports* issue and the *PACE Buying Guide*. Pickup trucks and Sport/Utility vehicles were eliminated due to incomplete information in the *Consumer Reports* source. Duplicate models (e.g., Dodge Shadow and Plymouth Sundance) were listed at most once.

Further description can be found in Lock (1993).

Source

Lock, R. H. (1993) 1993 New Car Data. *Journal of Statistics Education* **1**(1). doi:10.1080/10691898.1993.11910459

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS*. Fourth Edition. Springer.

`cats` *Anatomical Data from Domestic Cats*

Description

The heart and body weights of samples of male and female cats used for *digitalis* experiments. The cats were all adult, over 2 kg body weight.

Usage

```
cats
```

Format

This data frame contains the following columns:

`Sex` sex: Factor with levels `"F"` and `"M"`.

`Bwt` body weight in kg.

`Hwt` heart weight in g.

Source

R. A. Fisher (1947) The analysis of covariance method for the relation between a part and the whole, *Biometrics* **3**, 65–68.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

`cement` *Heat Evolved by Setting Cements*

Description

Experiment on the heat evolved in the setting of each of 13 cements.

Usage

```
cement
```

Format

`x1, x2, x3, x4` Proportions (%) of active ingredients.

`y` heat evolved in cals/gm.

Details

Thirteen samples of Portland cement were set. For each sample, the percentages of the four main chemical ingredients was accurately measured. While the cement was setting the amount of heat evolved was also measured.

Source

Woods, H., Steinour, H.H. and Starke, H.R. (1932) Effect of composition of Portland cement on heat evolved during hardening. *Industrial Engineering and Chemistry*, **24**, 1207–1214.

References

Hald, A. (1957) *Statistical Theory with Engineering Applications.* Wiley, New York.

Examples

```
lm(y ~ x1 + x2 + x3 + x4, cement)
```

`chem` *Copper in Wholemeal Flour*

Description

A numeric vector of 24 determinations of copper in wholemeal flour, in parts per million.

Usage

```
chem
```

Source

Analytical Methods Committee (1989) Robust statistics – how not to reject outliers. *The Analyst* **114**, 1693–1702.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

`con2tr` *Convert Lists to Data Frames for use by lattice*

Description

Convert lists to data frames for use by lattice.

Usage

```
con2tr(obj)
```

Arguments

`obj`	A list of components `x`, `y` and `z` as passed to `contour`.

Details

`con2tr` repeats the `x` and `y` components suitably to match the vector `z`.

Value

A data frame suitable for passing to lattice (formerly trellis) functions.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

confint-MASS *Confidence Intervals for Model Parameters*

Description

Computes confidence intervals for one or more parameters in a fitted model. Package **MASS** added methods for `glm` and `nls` fits. As fron R 4.4.0 these have been migrated to package **stats**.

It also adds a method for `polr` fits.

contr.sdif *Successive Differences Contrast Coding*

Description

A coding for factors based on successive differences.

Usage

```
contr.sdif(n, contrasts = TRUE, sparse = FALSE)
```

Arguments

`n`	The number of levels required.
`contrasts`	logical: Should there be `n - 1` columns orthogonal to the mean (the default) or `n` columns spanning the space?
`sparse`	logical. If true and the result would be sparse (only true for `contrasts = FALSE`), return a sparse matrix.

Details

The contrast coefficients are chosen so that the coded coefficients in a one-way layout are the differences between the means of the second and first levels, the third and second levels, and so on. This makes most sense for ordered factors, but does not assume that the levels are equally spaced.

Value

If `contrasts` is `TRUE`, a matrix with `n` rows and `n - 1` columns, and the `n` by `n` identity matrix if `contrasts` is `FALSE`.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth Edition, Springer.

See Also

`contr.treatment`, `contr.sum`, `contr.helmert`.

Examples

```
(A <- contr.sdif(6))
zapsmall(ginv(A))
```

---

coop *Co-operative Trial in Analytical Chemistry*

Description

Seven specimens were sent to 6 laboratories in 3 separate batches and each analysed for Analyte. Each analysis was duplicated.

Usage

```
coop
```

Format

This data frame contains the following columns:

`Lab` Laboratory, `L1`, `L2`, ..., `L6`.

`Spc` Specimen, `S1`, `S2`, ..., `S7`.

`Bat` Batch, `B1`, `B2`, `B3` (nested within `Spc/Lab`),

`Conc` Concentration of Analyte in g/kg.

Source

Analytical Methods Committee (1987) Recommendations for the conduct and interpretation of co-operative trials, *The Analyst* **112**, 679–686.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

chem, abbey.

corresp *Simple Correspondence Analysis*

Description

Find the principal canonical correlation and corresponding row- and column-scores from a correspondence analysis of a two-way contingency table.

Usage

```
corresp(x, ...)

## S3 method for class 'matrix'
corresp(x, nf = 1, ...)

## S3 method for class 'factor'
corresp(x, y, ...)

## S3 method for class 'data.frame'
corresp(x, ...)

## S3 method for class 'xtabs'
corresp(x, ...)

## S3 method for class 'formula'
corresp(formula, data, ...)
```

Arguments

`x, formula`	The function is generic, accepting various forms of the principal argument for specifying a two-way frequency table. Currently accepted forms are matrices, data frames (coerced to frequency tables), objects of class `"xtabs"` and formulae of the form `~ F1 + F2`, where `F1` and `F2` are factors.
`nf`	The number of factors to be computed. Note that although 1 is the most usual, one school of thought takes the first two singular vectors for a sort of biplot.
`y`	a second factor for a cross-classification.
`data`	an optional data frame, list or environment against which to preferentially resolve variables in the formula.
`...`	If the principal argument is a formula, a data frame may be specified as well from which variables in the formula are preferentially satisfied.

Details

See Venables & Ripley (2002). The `plot` method produces a graphical representation of the table if `nf=1`, with the *areas* of circles representing the numbers of points. If `nf` is two or more the `biplot` method is called, which plots the second and third columns of the matrices `A = Dr^(-1/2)`

U L and B = Dc^(-1/2) V L where the singular value decomposition is U L V. Thus the x-axis is the canonical correlation times the row and column scores. Although this is called a biplot, it does *not* have any useful inner product relationship between the row and column scores. Think of this as an equally-scaled plot with two unrelated sets of labels. The origin is marked on the plot with a cross. (For other versions of this plot see the book.)

Value

An list object of class "correspondence" for which print, plot and biplot methods are supplied. The main components are the canonical correlation(s) and the row and column scores.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

Gower, J. C. and Hand, D. J. (1996) *Biplots.* Chapman & Hall.

See Also

svd, princomp.

Examples

```
## IGNORE_RDIFF_BEGIN
## The signs can vary by platform
(ct <- corresp(~ Age + Eth, data = quine))
plot(ct)

corresp(caith)
biplot(corresp(caith, nf = 2))
## IGNORE_RDIFF_END
```

---

cov.rob *Resistant Estimation of Multivariate Location and Scatter*

Description

Compute a multivariate location and scale estimate with a high breakdown point – this can be thought of as estimating the mean and covariance of the good part of the data. cov.mve and cov.mcd are compatibility wrappers.

Usage

```
cov.rob(x, cor = FALSE, quantile.used = floor((n + p + 1)/2),
        method = c("mve", "mcd", "classical"),
        nsamp = "best", seed)

cov.mve(...)
cov.mcd(...)
```

Arguments

x	a matrix or data frame.
cor	should the returned result include a correlation matrix?
quantile.used	the minimum number of the data points regarded as good points.
method	the method to be used – minimum volume ellipsoid, minimum covariance determinant or classical product-moment. Using cov.mve or cov.mcd forces mve or mcd respectively.
nsamp	the number of samples or "best" or "exact" or "sample". The limit If "sample" the number chosen is min(5*p, 3000), taken from Rousseeuw and Hubert (1997). If "best" exhaustive enumeration is done up to 5000 samples: if "exact" exhaustive enumeration will be attempted.
seed	the seed to be used for random sampling: see RNGkind. The current value of .Random.seed will be preserved if it is set.
...	arguments to cov.rob other than method.

Details

For method "mve", an approximate search is made of a subset of size quantile.used with an enclosing ellipsoid of smallest volume; in method "mcd" it is the volume of the Gaussian confidence ellipsoid, equivalently the determinant of the classical covariance matrix, that is minimized. The mean of the subset provides a first estimate of the location, and the rescaled covariance matrix a first estimate of scatter. The Mahalanobis distances of all the points from the location estimate for this covariance matrix are calculated, and those points within the 97.5% point under Gaussian assumptions are declared to be good. The final estimates are the mean and rescaled covariance of the good points.

The rescaling is by the appropriate percentile under Gaussian data; in addition the first covariance matrix has an *ad hoc* finite-sample correction given by Marazzi.

For method "mve" the search is made over ellipsoids determined by the covariance matrix of p of the data points. For method "mcd" an additional improvement step suggested by Rousseeuw and van Driessen (1999) is used, in which once a subset of size quantile.used is selected, an ellipsoid based on its covariance is tested (as this will have no larger a determinant, and may be smaller).

There is a hard limit on the allowed number of samples, $2^{31} - 1$. However, practical limits are likely to be much lower and one might check the number of samples used for exhaustive enumeration, combn(NROW(x), NCOL(x) + 1), before attempting it.

Value

A list with components

center	the final estimate of location.
cov	the final estimate of scatter.
cor	(only is cor = TRUE) the estimate of the correlation matrix.
sing	message giving number of singular samples out of total
crit	the value of the criterion on log scale. For MCD this is the determinant, and for MVE it is proportional to the volume.

best	the subset used. For MVE the best sample, for MCD the best set of size quantile.used.
n.obs	total number of observations.

References

P. J. Rousseeuw and A. M. Leroy (1987) *Robust Regression and Outlier Detection.* Wiley.

A. Marazzi (1993) *Algorithms, Routines and S Functions for Robust Statistics.* Wadsworth and Brooks/Cole.

P. J. Rousseeuw and B. C. van Zomeren (1990) Unmasking multivariate outliers and leverage points, *Journal of the American Statistical Association*, **85**, 633–639.

P. J. Rousseeuw and K. van Driessen (1999) A fast algorithm for the minimum covariance determinant estimator. *Technometrics* **41**, 212–223.

P. Rousseeuw and M. Hubert (1997) Recent developments in PROGRESS. In *L1-Statistical Procedures and Related Topics* ed Y. Dodge, IMS Lecture Notes volume **31**, pp. 201–214.

See Also

lqs

Examples

```
set.seed(123)
cov.rob(stackloss)
cov.rob(stack.x, method = "mcd", nsamp = "exact")
```

---

cov.trob *Covariance Estimation for Multivariate t Distribution*

Description

Estimates a covariance or correlation matrix assuming the data came from a multivariate t distribution: this provides some degree of robustness to outlier without giving a high breakdown point.

Usage

```
cov.trob(x, wt = rep(1, n), cor = FALSE, center = TRUE, nu = 5,
         maxit = 25, tol = 0.01)
```

Arguments

x	data matrix. Missing values (NAs) are not allowed.
wt	A vector of weights for each case: these are treated as if the case i actually occurred wt[i] times.
cor	Flag to choose between returning the correlation (cor = TRUE) or covariance (cor = FALSE) matrix.

center	a logical value or a numeric vector providing the location about which the covariance is to be taken. If center = FALSE, no centering is done; if center = TRUE the MLE of the location vector is used.
nu	'degrees of freedom' for the multivariate t distribution. Must exceed 2 (so that the covariance matrix is finite).
maxit	Maximum number of iterations in fitting.
tol	Convergence tolerance for fitting.

Value

A list with the following components

cov	the fitted covariance matrix.
center	the estimated or specified location vector.
wt	the specified weights: only returned if the wt argument was given.
n.obs	the number of cases used in the fitting.
cor	the fitted correlation matrix: only returned if cor = TRUE.
call	The matched call.
iter	The number of iterations used.

References

J. T. Kent, D. E. Tyler and Y. Vardi (1994) A curious likelihood identity for the multivariate t-distribution. *Communications in Statistics—Simulation and Computation* **23**, 441–453.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS*. Fourth Edition. Springer.

See Also

cov, cov.wt, cov.mve

Examples

```
cov.trob(stackloss)
```

---

cpus *Performance of Computer CPUs*

Description

A relative performance measure and characteristics of 209 CPUs.

Usage

```
cpus
```

Format

The components are:

`name` manufacturer and model.

`syct` cycle time in nanoseconds.

`mmin` minimum main memory in kilobytes.

`mmax` maximum main memory in kilobytes.

`cach` cache size in kilobytes.

`chmin` minimum number of channels.

`chmax` maximum number of channels.

`perf` published performance on a benchmark mix relative to an IBM 370/158-3.

`estperf` estimated performance (by Ein-Dor & Feldmesser).

Source

P. Ein-Dor and J. Feldmesser (1987) Attributes of the performance of central processing units: a relative performance prediction model. *Comm. ACM.* **30**, 308–317.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

`crabs` *Morphological Measurements on Leptograpsus Crabs*

Description

The `crabs` data frame has 200 rows and 8 columns, describing 5 morphological measurements on 50 crabs each of two colour forms and both sexes, of the species *Leptograpsus variegatus* collected at Fremantle, W. Australia.

Usage

```
crabs
```

Format

This data frame contains the following columns:

`sp species` - `"B"` or `"O"` for blue or orange.

`sex` as it says.

`index` index `1:50` within each of the four groups.

`FL` frontal lobe size (mm).

`RW` rear width (mm).

`CL` carapace length (mm).

`CW` carapace width (mm).

`BD` body depth (mm).

Source

Campbell, N.A. and Mahon, R.J. (1974) A multivariate study of variation in two species of rock crab of genus *Leptograpsus. Australian Journal of Zoology* **22**, 417–425.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

`Cushings` *Diagnostic Tests on Patients with Cushing's Syndrome*

Description

Cushing's syndrome is a hypertensive disorder associated with over-secretion of cortisol by the adrenal gland. The observations are urinary excretion rates of two steroid metabolites.

Usage

```
Cushings
```

Format

The `Cushings` data frame has 27 rows and 3 columns:

`Tetrahydrocortisone` urinary excretion rate (mg/24hr) of Tetrahydrocortisone.

`Pregnanetriol` urinary excretion rate (mg/24hr) of Pregnanetriol.

`Type` underlying type of syndrome, coded `a` (adenoma) , `b` (bilateral hyperplasia), `c` (carcinoma) or `u` for unknown.

Source

J. Aitchison and I. R. Dunsmore (1975) *Statistical Prediction Analysis.* Cambridge University Press, Tables 11.1–3.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

DDT *DDT in Kale*

Description

A numeric vector of 15 measurements by different laboratories of the pesticide DDT in kale, in ppm (parts per million) using the multiple pesticide residue measurement.

Usage

```
DDT
```

Source

C. E. Finsterwalder (1976) Collaborative study of an extension of the Mills *et al* method for the determination of pesticide residues in food. *J. Off. Anal. Chem.* **59**, 169–171

R. G. Staudte and S. J. Sheather (1990) *Robust Estimation and Testing.* Wiley

deaths *Monthly Deaths from Lung Diseases in the UK*

Description

A time series giving the monthly deaths from bronchitis, emphysema and asthma in the UK, 1974-1979, both sexes (`deaths`),

Usage

```
deaths
```

Source

P. J. Diggle (1990) *Time Series: A Biostatistical Introduction.* Oxford, table A.3

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

See Also

This the same as dataset `ldeaths` in R's **datasets** package.

`denumerate` *Transform an Allowable Formula for 'loglm' into one for 'terms'*

Description

`loglm` allows dimension numbers to be used in place of names in the formula. `denumerate` modifies such a formula into one that `terms` can process.

Usage

```
denumerate(x)
```

Arguments

`x`	A formula conforming to the conventions of `loglm`, that is, it may allow dimension numbers to stand in for names when specifying a log-linear model.

Details

The model fitting function `loglm` fits log-linear models to frequency data using iterative proportional scaling. To specify the model the user must nominate the margins in the data that remain fixed under the log-linear model. It is convenient to allow the user to use dimension numbers, 1, 2, 3, ... for the first, second, third, ..., margins in a similar way to variable names. As the model formula has to be parsed by `terms`, which treats `1` in a special way and requires parseable variable names, these formulae have to be modified by giving genuine names for these margin, or dimension numbers. `denumerate` replaces these numbers with names of a special form, namely `n` is replaced by `.vn`. This allows `terms` to parse the formula in the usual way.

Value

A linear model formula like that presented, except that where dimension numbers, say `n`, have been used to specify fixed margins these are replaced by names of the form `.vn` which may be processed by `terms`.

See Also

`renumerate`

Examples

```
denumerate(~(1+2+3)^3 + a/b)
## which gives ~ (.v1 + .v2 + .v3)^3 + a/b
```

dose.p *Predict Doses for Binomial Assay model*

Description

Calibrate binomial assays, generalizing the calculation of LD50.

Usage

```
dose.p(obj, cf = 1:2, p = 0.5)
```

Arguments

obj	A fitted model object of class inheriting from "glm".
cf	The terms in the coefficient vector giving the intercept and coefficient of (log-)dose
p	Probabilities at which to predict the dose needed.

Value

An object of class "glm.dose" giving the prediction (attribute "p" and standard error (attribute "SE") at each response probability.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Springer.

Examples

```
ldose <- rep(0:5, 2)
numdead <- c(1, 4, 9, 13, 18, 20, 0, 2, 6, 10, 12, 16)
sex <- factor(rep(c("M", "F"), c(6, 6)))
SF <- cbind(numdead, numalive = 20 - numdead)
budworm.lg0 <- glm(SF ~ sex + ldose - 1, family = binomial)

dose.p(budworm.lg0, cf = c(1,3), p = 1:3/4)
dose.p(update(budworm.lg0, family = binomial(link=probit)),
       cf = c(1,3), p = 1:3/4)
```

drivers *Deaths of Car Drivers in Great Britain 1969-84*

Description

A regular time series giving the monthly totals of car drivers in Great Britain killed or seriously injured Jan 1969 to Dec 1984. Compulsory wearing of seat belts was introduced on 31 Jan 1983

Usage

```
drivers
```

Source

Harvey, A.C. (1989) *Forecasting, Structural Time Series Models and the Kalman Filter.* Cambridge University Press, pp. 519–523.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

dropterm *Try All One-Term Deletions from a Model*

Description

Try fitting all models that differ from the current model by dropping a single term, maintaining marginality.

This function is generic; there exist methods for classes `lm` and `glm` and the default method will work for many other classes.

Usage

```
dropterm (object, ...)

## Default S3 method:
dropterm(object, scope, scale = 0, test = c("none", "Chisq"),
         k = 2, sorted = FALSE, trace = FALSE, ...)

## S3 method for class 'lm'
dropterm(object, scope, scale = 0, test = c("none", "Chisq", "F"),
         k = 2, sorted = FALSE, ...)

## S3 method for class 'glm'
dropterm(object, scope, scale = 0, test = c("none", "Chisq", "F"),
         k = 2, sorted = FALSE, trace = FALSE, ...)
```

Arguments

object	A object fitted by some model-fitting function.
scope	a formula giving terms which might be dropped. By default, the model formula. Only terms that can be dropped and maintain marginality are actually tried.
scale	used in the definition of the AIC statistic for selecting the models, currently only for lm, aov and glm models. Specifying scale asserts that the residual standard error or dispersion is known.
test	should the results include a test statistic relative to the original model? The F test is only appropriate for lm and aov models, and perhaps for some over-dispersed glm models. The Chisq test can be an exact test (lm models with known scale) or a likelihood-ratio test depending on the method.
k	the multiple of the number of degrees of freedom used for the penalty. Only k = 2 gives the genuine AIC: k = log(n) is sometimes referred to as BIC or SBC.
sorted	should the results be sorted on the value of AIC?
trace	if TRUE additional information may be given on the fits as they are tried.
...	arguments passed to or from other methods.

Details

The definition of AIC is only up to an additive constant: when appropriate (lm models with specified scale) the constant is taken to be that used in Mallows' Cp statistic and the results are labelled accordingly.

Value

A table of class "anova" containing at least columns for the change in degrees of freedom and AIC (or Cp) for the models. Some methods will give further information, for example sums of squares, deviances, log-likelihoods and test statistics.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

addterm, stepAIC

Examples

```
quine.hi <- aov(log(Days + 2.5) ~ .^4, quine)
quine.nxt <- update(quine.hi, . ~ . - Eth:Sex:Age:Lrn)
dropterm(quine.nxt, test=  "F")
quine.stp <- stepAIC(quine.nxt,
    scope = list(upper = ~Eth*Sex*Age*Lrn, lower = ~1),
    trace = FALSE)
dropterm(quine.stp, test = "F")
quine.3 <- update(quine.stp, . ~ . - Eth:Age:Lrn)
dropterm(quine.3, test = "F")
```

```
quine.4 <- update(quine.3, . ~ . - Eth:Age)
dropterm(quine.4, test = "F")
quine.5 <- update(quine.4, . ~ . - Age:Lrn)
dropterm(quine.5, test = "F")

house.glm0 <- glm(Freq ~ Infl*Type*Cont + Sat, family=poisson,
                  data = housing)
house.glm1 <- update(house.glm0, . ~ . + Sat*(Infl+Type+Cont))
dropterm(house.glm1, test = "Chisq")
```

`eagles` *Foraging Ecology of Bald Eagles*

Description

Knight and Skagen collected during a field study on the foraging behaviour of wintering Bald Eagles in Washington State, USA data concerning 160 attempts by one (pirating) Bald Eagle to steal a chum salmon from another (feeding) Bald Eagle.

Usage

```
eagles
```

Format

The `eagles` data frame has 8 rows and 5 columns.

- `y` Number of successful attempts.
- `n` Total number of attempts.
- `P` Size of pirating eagle (`L` = large, `S` = small).
- `A` Age of pirating eagle (`I` = immature, `A` = adult).
- `V` Size of victim eagle (`L` = large, `S` = small).

Source

Knight, R. L. and Skagen, S. K. (1988) Agonistic asymmetries and the foraging ecology of Bald Eagles. *Ecology* **69**, 1188–1194.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

Examples

```
eagles.glm <- glm(cbind(y, n - y) ~ P*A + V, data = eagles,
                  family = binomial)
dropterm(eagles.glm)
prof <- profile(eagles.glm)
plot(prof)
pairs(prof)
```

`epil` *Seizure Counts for Epileptics*

Description

Thall and Vail (1990) give a data set on two-week seizure counts for 59 epileptics. The number of seizures was recorded for a baseline period of 8 weeks, and then patients were randomly assigned to a treatment group or a control group. Counts were then recorded for four successive two-week periods. The subject's age is the only covariate.

Usage

```
epil
```

Format

This data frame has 236 rows and the following 9 columns:

`y` the count for the 2-week period.

`trt` treatment, `"placebo"` or `"progabide"`.

`base` the counts in the baseline 8-week period.

`age` subject's age, in years.

`V4` `0/1` indicator variable of period 4.

`subject` subject number, 1 to 59.

`period` period, 1 to 4.

`lbase` log-counts for the baseline period, centred to have zero mean.

`lage` log-ages, centred to have zero mean.

Source

Thall, P. F. and Vail, S. C. (1990) Some covariance models for longitudinal count data with over-dispersion. *Biometrics* **46**, 657–671.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth Edition. Springer.

Examples

```
## IGNORE_RDIFF_BEGIN
summary(glm(y ~ lbase*trt + lage + V4, family = poisson,
            data = epil), correlation = FALSE)
## IGNORE_RDIFF_END
epil2 <- epil[epil$period == 1, ]
epil2["period"] <- rep(0, 59); epil2["y"] <- epil2["base"]
epil["time"] <- 1; epil2["time"] <- 4
epil2 <- rbind(epil, epil2)
epil2$pred <- unclass(epil2$trt) * (epil2$period > 0)
epil2$subject <- factor(epil2$subject)
epil3 <- aggregate(epil2, list(epil2$subject, epil2$period > 0),
   function(x) if(is.numeric(x)) sum(x) else x[1])
epil3$pred <- factor(epil3$pred,
   labels = c("base", "placebo", "drug"))

contrasts(epil3$pred) <- structure(contr.sdif(3),
    dimnames = list(NULL, c("placebo-base", "drug-placebo")))
## IGNORE_RDIFF_BEGIN
summary(glm(y ~ pred + factor(subject) + offset(log(time)),
            family = poisson, data = epil3), correlation = FALSE)
## IGNORE_RDIFF_END

summary(glmmPQL(y ~ lbase*trt + lage + V4,
                random = ~ 1 | subject,
                family = poisson, data = epil))
summary(glmmPQL(y ~ pred, random = ~1 | subject,
                family = poisson, data = epil3))
```

`eqscplot` *Plots with Geometrically Equal Scales*

Description

Version of a scatterplot with scales chosen to be equal on both axes, that is 1cm represents the same units on each

Usage

```
eqscplot(x, y, ratio = 1, tol = 0.04, uin, ...)
```

Arguments

`x`	vector of x values, or a 2-column matrix, or a list with components `x` and `y`
`y`	vector of y values
`ratio`	desired ratio of units on the axes. Units on the y axis are drawn at `ratio` times the size of units on the x axis. Ignored if `uin` is specified and of length 2.
`tol`	proportion of white space at the margins of plot

`uin`	desired values for the units-per-inch parameter. If of length 1, the desired units per inch on the x axis.
`...`	further arguments for `plot` and graphical parameters. Note that `par(xaxs="i", yaxs="i")` is enforced, and `xlim` and `ylim` will be adjusted accordingly.

Details

Limits for the x and y axes are chosen so that they include the data. One of the sets of limits is then stretched from the midpoint to make the units in the ratio given by `ratio`. Finally both are stretched by `1 + tol` to move points away from the axes, and the points plotted.

Value

invisibly, the values of `uin` used for the plot.

Side Effects

performs the plot.

Note

Arguments `ratio` and `uin` were suggested by Bill Dunlap.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

`plot`, `par`

`farms` *Ecological Factors in Farm Management*

Description

The `farms` data frame has 20 rows and 4 columns. The rows are farms on the Dutch island of Terschelling and the columns are factors describing the management of grassland.

Usage

```
farms
```

Format

This data frame contains the following columns:

`Mois` Five levels of soil moisture – level 3 does not occur at these 20 farms.

`Manag` Grassland management type (`SF` = standard, `BF` = biological, `HF` = hobby farming, `NM` = nature conservation).

`Use` Grassland use (`U1` = hay production, `U2` = intermediate, `U3` = grazing).

`Manure` Manure usage – classes `C0` to `C4`.

Source

J.C. Gower and D.J. Hand (1996) *Biplots*. Chapman & Hall, Table 4.6.

Quoted as from:
R.H.G. Jongman, C.J.F. ter Braak and O.F.R. van Tongeren (1987) *Data Analysis in Community and Landscape Ecology.* PUDOC, Wageningen.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

Examples

```
farms.mca <- mca(farms, abbrev = TRUE)  # Use levels as names
eqscplot(farms.mca$cs, type = "n")
text(farms.mca$rs, cex = 0.7)
text(farms.mca$cs, labels = dimnames(farms.mca$cs)[[1]], cex = 0.7)
```

---

fgl *Measurements of Forensic Glass Fragments*

Description

The `fgl` data frame has 214 rows and 10 columns. It was collected by B. German on fragments of glass collected in forensic work.

Usage

```
fgl
```

Format

This data frame contains the following columns:

`RI` refractive index; more precisely the refractive index is 1.518xxxx.

The next 8 measurements are percentages by weight of oxides.

`Na` sodium.

`Mg` manganese.

`Al` aluminium.

`Si` silicon.

`K` potassium.

`Ca` calcium.

`Ba` barium.

`Fe` iron.

`type` The fragments were originally classed into seven types, one of which was absent in this dataset. The categories which occur are window float glass (`WinF`: 70), window non-float glass (`WinNF`: 76), vehicle window glass (`Veh`: 17), containers (`Con`: 13), tableware (`Tabl`: 9) and vehicle headlamps (`Head`: 29).

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

`fitdistr` *Maximum-likelihood Fitting of Univariate Distributions*

Description

Maximum-likelihood fitting of univariate distributions, allowing parameters to be held fixed if desired.

Usage

```
fitdistr(x, densfun, start, ...)
```

Arguments

`x`	A numeric vector of length at least one containing only finite values.
`densfun`	Either a character string or a function returning a density evaluated at its first argument. Distributions `"beta"`, `"cauchy"`, `"chi-squared"`, `"exponential"`, `"gamma"`, `"geometric"`, `"log-normal"`, `"lognormal"`, `"logistic"`, `"negative binomial"`, `"normal"`, `"Poisson"`, `"t"` and `"weibull"` are recognised, case being ignored.
`start`	A named list giving the parameters to be optimized with initial values. This can be omitted for some of the named distributions and must be for others (see Details).
`...`	Additional parameters, either for `densfun` or for `optim`. In particular, it can be used to specify bounds via `lower` or `upper` or both. If arguments of `densfun` (or the density function corresponding to a character-string specification) are included they will be held fixed.

Details

For the Normal, log-Normal, geometric, exponential and Poisson distributions the closed-form MLEs (and exact standard errors) are used, and `start` should not be supplied.

For all other distributions, direct optimization of the log-likelihood is performed using `optim`. The estimated standard errors are taken from the observed information matrix, calculated by a numerical approximation. For one-dimensional problems the Nelder-Mead method is used and for multi-dimensional problems the BFGS method, unless arguments named `lower` or `upper` are supplied (when `L-BFGS-B` is used) or `method` is supplied explicitly.

For the `"t"` named distribution the density is taken to be the location-scale family with location `m` and scale `s`.

For the following named distributions, reasonable starting values will be computed if `start` is omitted or only partially specified: `"cauchy"`, `"gamma"`, `"logistic"`, `"negative binomial"` (parametrized by `mu` and `size`), `"t"` and `"weibull"`. Note that these starting values may not be good enough if the fit is poor: in particular they are not resistant to outliers unless the fitted distribution is long-tailed.

There are `print`, `coef`, `vcov` and `logLik` methods for class `"fitdistr"`.

Value

An object of class `"fitdistr"`, a list with four components,

`estimate`	the parameter estimates,
`sd`	the estimated standard errors,
`vcov`	the estimated variance-covariance matrix, and
`loglik`	the log-likelihood.

Note

Numerical optimization cannot work miracles: please note the comments in `optim` on scaling data. If the fitted parameters are far away from one, consider re-fitting specifying the control parameter `parscale`.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

Examples

```
## avoid spurious accuracy
op <- options(digits = 3)
set.seed(123)
x <- rgamma(100, shape = 5, rate = 0.1)
fitdistr(x, "gamma")
## now do this directly with more control.
fitdistr(x, dgamma, list(shape = 1, rate = 0.1), lower = 0.001)

set.seed(123)
x2 <- rt(250, df = 9)
fitdistr(x2, "t", df = 9)
```

```
## allow df to vary: not a very good idea!
fitdistr(x2, "t")
## now do fixed-df fit directly with more control.
mydt <- function(x, m, s, df) dt((x-m)/s, df)/s
fitdistr(x2, mydt, list(m = 0, s = 1), df = 9, lower = c(-Inf, 0))

set.seed(123)
x3 <- rweibull(100, shape = 4, scale = 100)
fitdistr(x3, "weibull")

set.seed(123)
x4 <- rnegbin(500, mu = 5, theta = 4)
fitdistr(x4, "Negative Binomial")
options(op)
```

`forbes` *Forbes' Data on Boiling Points in the Alps*

Description

A data frame with 17 observations on boiling point of water and barometric pressure in inches of mercury.

Usage

```
forbes
```

Format

`bp` boiling point (degrees Farenheit).

`pres` barometric pressure in inches of mercury.

Source

A. C. Atkinson (1985) *Plots, Transformations and Regression.* Oxford.

S. Weisberg (1980) *Applied Linear Regression.* Wiley.

fractions *Rational Approximation*

Description

Find rational approximations to the components of a real numeric object using a standard continued fraction method.

Usage

```
fractions(x, cycles = 10, max.denominator = 2000, ...)

as.fractions(x)

is.fractions(f)
```

Arguments

x	Any object of mode numeric. Missing values are now allowed.
cycles	The maximum number of steps to be used in the continued fraction approximation process.
max.denominator	An early termination criterion. If any partial denominator exceeds max.denominator the continued fraction stops at that point.
...	arguments passed to or from other methods.
f	an R object.

Details

Each component is first expanded in a continued fraction of the form

x = floor(x) + 1/(p1 + 1/(p2 + ...)))

where p1, p2, ... are positive integers, terminating either at cycles terms or when a pj > max.denominator. The continued fraction is then re-arranged to retrieve the numerator and denominator as integers.

The numerators and denominators are then combined into a character vector that becomes the "fracs" attribute and used in printed representations.

Arithmetic operations on "fractions" objects have full floating point accuracy, but the character representation printed out may not.

Value

An object of class "fractions". A structure with .Data component the same as the input numeric x, but with the rational approximations held as a character vector attribute, "fracs". Arithmetic operations on "fractions" objects are possible.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth Edition. Springer.

See Also

rational

Examples

```
X <- matrix(runif(25), 5, 5)
zapsmall(solve(X, X/5)) # print near-zeroes as zero
fractions(solve(X, X/5))
fractions(solve(X, X/5)) + 1
```

GAGurine *Level of GAG in Urine of Children*

Description

Data were collected on the concentration of a chemical GAG in the urine of 314 children aged from zero to seventeen years. The aim of the study was to produce a chart to help a paediatrican to assess if a child's GAG concentration is 'normal'.

Usage

```
GAGurine
```

Format

This data frame contains the following columns:

`Age` age of child in years.

`GAG` concentration of GAG (the units have been lost).

Source

Mrs Susan Prosser, Paediatrics Department, University of Oxford, via Department of Statistics Consulting Service.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

`galaxies` *Velocities for 82 Galaxies*

Description

A numeric vector of velocities in km/sec of 82 galaxies from 6 well-separated conic sections of an `unfilled` survey of the Corona Borealis region. Multimodality in such surveys is evidence for voids and superclusters in the far universe.

Usage

```
galaxies
```

Note

There is an 83rd measurement of 5607 km/sec in the Postman *et al.* paper which is omitted in Roeder (1990) and from the dataset here.

There is also a typo: this dataset has 78th observation 26690 which should be 26960.

Source

Roeder, K. (1990) Density estimation with confidence sets exemplified by superclusters and voids in galaxies. *Journal of the American Statistical Association* **85**, 617–624.

Postman, M., Huchra, J. P. and Geller, M. J. (1986) Probes of large-scale structures in the Corona Borealis region. *Astronomical Journal* **92**, 1238–1247.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

Examples

```
gal <- galaxies/1000
c(width.SJ(gal, method = "dpi"), width.SJ(gal))
plot(x = c(0, 40), y = c(0, 0.3), type = "n", bty = "l",
     xlab = "velocity of galaxy (1000km/s)", ylab = "density")
rug(gal)
lines(density(gal, width = 3.25, n = 200), lty = 1)
lines(density(gal, width = 2.56, n = 200), lty = 3)
```

gamma.dispersion *Calculate the MLE of the Gamma Dispersion Parameter in a GLM Fit*

Description

A front end to gamma.shape for convenience. Finds the reciprocal of the estimate of the shape parameter only.

Usage

```
gamma.dispersion(object, ...)
```

Arguments

object	Fitted model object giving the gamma fit.
...	Additional arguments passed on to gamma.shape.

Value

The MLE of the dispersion parameter of the gamma distribution.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

gamma.shape.glm, including the example on its help page.

gamma.shape *Estimate the Shape Parameter of the Gamma Distribution in a GLM Fit*

Description

Find the maximum likelihood estimate of the shape parameter of the gamma distribution after fitting a Gamma generalized linear model.

Usage

```
gamma.shape(object, ...)

## S3 method for class 'glm'
gamma.shape(object, it.lim = 10,
            eps.max = .Machine$double.eps^0.25, verbose = FALSE, ...)
```

Arguments

object	Fitted model object from a Gamma family or quasi family with variance = "mu^2".
it.lim	Upper limit on the number of iterations.
eps.max	Maximum discrepancy between approximations for the iteration process to continue.
verbose	If TRUE, causes successive iterations to be printed out. The initial estimate is taken from the deviance.
...	further arguments passed to or from other methods.

Details

A glm fit for a Gamma family correctly calculates the maximum likelihood estimate of the mean parameters but provides only a crude estimate of the dispersion parameter. This function takes the results of the glm fit and solves the maximum likelihood equation for the reciprocal of the dispersion parameter, which is usually called the shape (or exponent) parameter.

Value

List of two components

alpha	the maximum likelihood estimate
SE	the approximate standard error, the square-root of the reciprocal of the observed information.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

gamma.dispersion

Examples

```
clotting <- data.frame(
    u = c(5,10,15,20,30,40,60,80,100),
    lot1 = c(118,58,42,35,27,25,21,19,18),
    lot2 = c(69,35,26,21,18,16,13,12,12))
clot1 <- glm(lot1 ~ log(u), data = clotting, family = Gamma)
gamma.shape(clot1)

gm <- glm(Days + 0.1 ~ Age*Eth*Sex*Lrn,
          quasi(link=log, variance="mu^2"), quine,
          start = c(3, rep(0,31)))
gamma.shape(gm, verbose = TRUE)
## IGNORE_RDIFF_BEGIN
summary(gm, dispersion = gamma.dispersion(gm))  # better summary
## IGNORE_RDIFF_END
```

gehan *Remission Times of Leukaemia Patients*

Description

A data frame from a trial of 42 leukaemia patients. Some were treated with the drug *6-mercaptopurine* and the rest are controls. The trial was designed as matched pairs, both withdrawn from the trial when either came out of remission.

Usage

```
gehan
```

Format

This data frame contains the following columns:

`pair` label for pair.

`time` remission time in weeks.

`cens` censoring, 0/1.

`treat` treatment, control or 6-MP.

Source

Cox, D. R. and Oakes, D. (1984) *Analysis of Survival Data.* Chapman & Hall, p. 7. Taken from

Gehan, E.A. (1965) A generalized Wilcoxon test for comparing arbitrarily single-censored samples. *Biometrika* **52**, 203–233.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

Examples

```
library(survival)
gehan.surv <- survfit(Surv(time, cens) ~ treat, data = gehan,
     conf.type = "log-log")
summary(gehan.surv)
survreg(Surv(time, cens) ~ factor(pair) + treat, gehan, dist = "exponential")
summary(survreg(Surv(time, cens) ~ treat, gehan, dist = "exponential"))
summary(survreg(Surv(time, cens) ~ treat, gehan))
gehan.cox <- coxph(Surv(time, cens) ~ treat, gehan)
summary(gehan.cox)
```

genotype *Rat Genotype Data*

Description

Data from a foster feeding experiment with rat mothers and litters of four different genotypes: `A`, `B`, `I` and `J`. Rat litters were separated from their natural mothers at birth and given to foster mothers to rear.

Usage

```
genotype
```

Format

The data frame has the following components:

`Litter` genotype of the litter.

`Mother` genotype of the foster mother.

`Wt` Litter average weight gain of the litter, in grams at age 28 days. (The source states that the within-litter variability is negligible.)

Source

Scheffe, H. (1959) *The Analysis of Variance* Wiley p. 140.

Bailey, D. W. (1953) *The Inheritance of Maternal Influences on the Growth of the Rat.* Unpublished Ph.D. thesis, University of California. Table B of the Appendix.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

geyser *Old Faithful Geyser Data*

Description

A version of the eruptions data from the 'Old Faithful' geyser in Yellowstone National Park, Wyoming. This version comes from Azzalini and Bowman (1990) and is of continuous measurement from August 1 to August 15, 1985.

Some nocturnal duration measurements were coded as 2, 3 or 4 minutes, having originally been described as 'short', 'medium' or 'long'.

Usage

```
geyser
```

Format

A data frame with 299 observations on 2 variables.

`duration`	numeric	Eruption time in mins
`waiting`	numeric	Waiting time for this eruption

Note

The `waiting` time was incorrectly described as the time to the next eruption in the original files, and corrected for **MASS** version 7.3-30.

References

Azzalini, A. and Bowman, A. W. (1990) A look at some data on the Old Faithful geyser. *Applied Statistics* **39**, 357–365.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

`faithful`.

CRAN package **sm**.

`gilgais` *Line Transect of Soil in Gilgai Territory*

Description

This dataset was collected on a line transect survey in gilgai territory in New South Wales, Australia. Gilgais are natural gentle depressions in otherwise flat land, and sometimes seem to be regularly distributed. The data collection was stimulated by the question: are these patterns reflected in soil properties? At each of 365 sampling locations on a linear grid of 4 meters spacing, samples were taken at depths 0-10 cm, 30-40 cm and 80-90 cm below the surface. pH, electrical conductivity and chloride content were measured on a 1:5 soil:water extract from each sample.

Usage

```
gilgais
```

Format

This data frame contains the following columns:

`pH00` pH at depth 0–10 cm.

`pH30` pH at depth 30–40 cm.

`pH80` pH at depth 80–90 cm.

`e00` electrical conductivity in mS/cm (0–10 cm).

`e30` electrical conductivity in mS/cm (30–40 cm).

`e80` electrical conductivity in mS/cm (80–90 cm).

`c00` chloride content in ppm (0–10 cm).

`c30` chloride content in ppm (30–40 cm).

`c80` chloride content in ppm (80–90 cm).

Source

Webster, R. (1977) Spectral analysis of gilgai soil. *Australian Journal of Soil Research* **15**, 191–204.

Laslett, G. M. (1989) Kriging and splines: An empirical comparison of their predictive performance in some applications (with discussion). *Journal of the American Statistical Association* **89**, 319–409

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

---

ginv *Generalized Inverse of a Matrix*

Description

Calculates the Moore-Penrose generalized inverse of a matrix `X`.

Usage

```
ginv(X, tol = sqrt(.Machine$double.eps))
```

Arguments

`X`	Matrix for which the Moore-Penrose inverse is required.
`tol`	A relative tolerance to detect zero singular values.

Value

A MP generalized inverse matrix for `X`.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS*. Fourth Edition. Springer.

See Also

`solve`, `svd`, `eigen`

`glm.convert` *Change a Negative Binomial fit to a GLM fit*

Description

This function modifies an output object from `glm.nb()` to one that looks like the output from `glm()` with a negative binomial family. This allows it to be updated keeping the theta parameter fixed.

Usage

```
glm.convert(object)
```

Arguments

`object`	An object of class `"negbin"`, typically the output from `glm.nb()`.

Details

Convenience function needed to effect some low level changes to the structure of the fitted model object.

Value

An object of class `"glm"` with negative binomial family. The theta parameter is then fixed at its present estimate.

See Also

`glm.nb`, `negative.binomial`, `glm`

Examples

```
quine.nb1 <- glm.nb(Days ~ Sex/(Age + Eth*Lrn), data = quine)
quine.nbA <- glm.convert(quine.nb1)
quine.nbB <- update(quine.nb1, . ~ . + Sex:Age:Lrn)
anova(quine.nbA, quine.nbB)
```

glm.nb *Fit a Negative Binomial Generalized Linear Model*

Description

A modification of the system function glm() to include estimation of the additional parameter, theta, for a Negative Binomial generalized linear model.

Usage

```
glm.nb(formula, data, weights, subset, na.action,
       start = NULL, etastart, mustart,
       control = glm.control(...), method = "glm.fit",
       model = TRUE, x = FALSE, y = TRUE, contrasts = NULL, ...,
       init.theta, link = log)
```

Arguments

formula, data, weights, subset, na.action, start, etastart, mustart, control, method, model, x, y, contras	arguments for the glm() function. Note that these exclude family and offset (but offset() can be used).
init.theta	Optional initial value for the theta parameter. If omitted a moment estimator after an initial fit using a Poisson GLM is used.
link	The link function. Currently must be one of log, sqrt or identity.

Details

An alternating iteration process is used. For given theta the GLM is fitted using the same process as used by glm(). For fixed means the theta parameter is estimated using score and information iterations. The two are alternated until convergence of both. (The number of alternations and the number of iterations when estimating theta are controlled by the maxit parameter of glm.control.)

Setting trace > 0 traces the alternating iteration process. Setting trace > 1 traces the glm fit, and setting trace > 2 traces the estimation of theta.

Value

A fitted model object of class negbin inheriting from glm and lm. The object is like the output of glm but contains three additional components, namely theta for the ML estimate of theta, SE.theta for its approximate standard error (using observed rather than expected information), and twologlik for twice the log-likelihood function.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

glm, negative.binomial, anova.negbin, summary.negbin, theta.md

There is a simulate method.

Examples

```
quine.nb1 <- glm.nb(Days ~ Sex/(Age + Eth*Lrn), data = quine)
quine.nb2 <- update(quine.nb1, . ~ . + Sex:Age:Lrn)
quine.nb3 <- update(quine.nb2, Days ~ .^4)
anova(quine.nb1, quine.nb2, quine.nb3)
```

---

glmmPQL *Fit Generalized Linear Mixed Models via PQL*

Description

Fit a GLMM model with multivariate normal random effects, using Penalized Quasi-Likelihood.

Usage

```
glmmPQL(fixed, random, family, data, correlation, weights,
        control, niter = 10, verbose = TRUE, ...)
```

Arguments

fixed	a two-sided linear formula giving fixed-effects part of the model.
random	a formula or list of formulae describing the random effects.
family	a GLM family.
data	an optional data frame, list or environment used as the first place to find variables in the formulae, weights and if present in ..., subset.
correlation	an optional correlation structure.
weights	optional case weights as in glm.
control	an optional argument to be passed to lme.
niter	maximum number of iterations.
verbose	logical: print out record of iterations?
...	Further arguments for lme.

Details

glmmPQL works by repeated calls to lme, so namespace **nlme** will be loaded at first use. (Before 2015 it used to attach nlme but nowadays only loads the namespace.)

Unlike lme, offset terms are allowed in fixed – this is done by pre- and post-processing the calls to lme.

Note that the returned object inherits from class "lme" and that most generics will use the method for that class. As from version 3.1-158, the fitted values have any offset included, as do the results of calling predict.

Value

A object of class `c("glmmPQL", "lme")`: see `lmeObject`.

References

Schall, R. (1991) Estimation in generalized linear models with random effects. *Biometrika* **78**, 719–727.

Breslow, N. E. and Clayton, D. G. (1993) Approximate inference in generalized linear mixed models. *Journal of the American Statistical Association* **88**, 9–25.

Wolfinger, R. and O'Connell, M. (1993) Generalized linear mixed models: a pseudo-likelihood approach. *Journal of Statistical Computation and Simulation* **48**, 233–243.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

`lme`

Examples

```
summary(glmmPQL(y ~ trt + I(week > 2), random = ~ 1 | ID,
                family = binomial, data = bacteria))

## an example of an offset: the coefficient of 'week' changes by one.
summary(glmmPQL(y ~ trt + week, random = ~ 1 | ID,
               family = binomial, data = bacteria))
summary(glmmPQL(y ~ trt + week + offset(week), random = ~ 1 | ID,
                family = binomial, data = bacteria))
```

---

hills *Record Times in Scottish Hill Races*

Description

The record times in 1984 for 35 Scottish hill races.

Usage

```
hills
```

Format

The components are:

`dist` distance in miles (on the map).

`climb` total height gained during the route, in feet.

`time` record time in minutes.

Source

A.C. Atkinson (1986) Comment: Aspects of diagnostic regression analysis. *Statistical Science* **1**, 397–402.

[A.C. Atkinson (1988) Transformations unmasked. *Technometrics* **30**, 311–318 "corrects" the time for Knock Hill from 78.65 to 18.65. It is unclear if this based on the original records.]

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

`hist.scott` *Plot a Histogram with Automatic Bin Width Selection*

Description

Plot a histogram with automatic bin width selection, using the Scott or Freedman–Diaconis formulae.

Usage

```
hist.scott(x, prob = TRUE, xlab = deparse(substitute(x)), ...)
hist.FD(x, prob = TRUE, xlab = deparse(substitute(x)), ...)
```

Arguments

`x`	A data vector
`prob`	Should the plot have unit area, so be a density estimate?
`xlab, ...`	Further arguments to `hist`.

Value

For the `nclass.*` functions, the suggested number of classes.

Side Effects

Plot a histogram.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Springer.

See Also

`hist`

housing *Frequency Table from a Copenhagen Housing Conditions Survey*

Description

The housing data frame has 72 rows and 5 variables.

Usage

```
housing
```

Format

`Sat` Satisfaction of householders with their present housing circumstances, (High, Medium or Low, ordered factor).

`Infl` Perceived degree of influence householders have on the management of the property (High, Medium, Low).

`Type` Type of rental accommodation, (Tower, Atrium, Apartment, Terrace).

`Cont` Contact residents are afforded with other residents, (Low, High).

`Freq` Frequencies: the numbers of residents in each class.

Source

Madsen, M. (1976) Statistical analysis of multiple contingency tables. Two examples. *Scand. J. Statist.* **3**, 97–106.

Cox, D. R. and Snell, E. J. (1984) *Applied Statistics, Principles and Examples*. Chapman & Hall.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

Examples

```
options(contrasts = c("contr.treatment", "contr.poly"))

# Surrogate Poisson models
house.glm0 <- glm(Freq ~ Infl*Type*Cont + Sat, family = poisson,
                  data = housing)
## IGNORE_RDIFF_BEGIN
summary(house.glm0, correlation = FALSE)
## IGNORE_RDIFF_END

addterm(house.glm0, ~. + Sat:(Infl+Type+Cont), test = "Chisq")

house.glm1 <- update(house.glm0, . ~ . + Sat*(Infl+Type+Cont))
## IGNORE_RDIFF_BEGIN
summary(house.glm1, correlation = FALSE)
```

```
## IGNORE_RDIFF_END

1 - pchisq(deviance(house.glm1), house.glm1$df.residual)

dropterm(house.glm1, test = "Chisq")

addterm(house.glm1, ~. + Sat:(Infl+Type+Cont)^2, test  =  "Chisq")

hnames <- lapply(housing[, -5], levels) # omit Freq
newData <- expand.grid(hnames)
newData$Sat <- ordered(newData$Sat)
house.pm <- predict(house.glm1, newData,
                    type = "response")  # poisson means
house.pm <- matrix(house.pm, ncol = 3, byrow = TRUE,
                   dimnames = list(NULL, hnames[[1]]))
house.pr <- house.pm/drop(house.pm %*% rep(1, 3))
cbind(expand.grid(hnames[-1]), round(house.pr, 2))

# Iterative proportional scaling
loglm(Freq ~ Infl*Type*Cont + Sat*(Infl+Type+Cont), data = housing)


# multinomial model
library(nnet)
(house.mult<- multinom(Sat ~ Infl + Type + Cont, weights = Freq,
                       data = housing))
house.mult2 <- multinom(Sat ~ Infl*Type*Cont, weights = Freq,
                        data = housing)
anova(house.mult, house.mult2)

house.pm <- predict(house.mult, expand.grid(hnames[-1]), type = "probs")
cbind(expand.grid(hnames[-1]), round(house.pm, 2))

# proportional odds model
house.cpr <- apply(house.pr, 1, cumsum)
logit <- function(x) log(x/(1-x))
house.ld <- logit(house.cpr[2, ]) - logit(house.cpr[1, ])
(ratio <- sort(drop(house.ld)))
mean(ratio)

(house.plr <- polr(Sat ~ Infl + Type + Cont,
                   data = housing, weights = Freq))

house.pr1 <- predict(house.plr, expand.grid(hnames[-1]), type = "probs")
cbind(expand.grid(hnames[-1]), round(house.pr1, 2))

Fr <- matrix(housing$Freq, ncol  =  3, byrow = TRUE)
2*sum(Fr*log(house.pr/house.pr1))

house.plr2 <- stepAIC(house.plr, ~.^2)
house.plr2$anova
```

huber *Huber M-estimator of Location with MAD Scale*

Description

Finds the Huber M-estimator of location with MAD scale.

Usage

```
huber(y, k = 1.5, tol = 1e-06)
```

Arguments

y	vector of data values
k	Winsorizes at k standard deviations
tol	convergence tolerance

Value

list of location and scale parameters

mu	location estimate
s	MAD scale estimate

References

Huber, P. J. (1981) *Robust Statistics.* Wiley.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

See Also

hubers, mad

Examples

```
huber(chem)
```

`hubers` *Huber Proposal 2 Robust Estimator of Location and/or Scale*

Description

Finds the Huber M-estimator for location with scale specified, scale with location specified, or both if neither is specified.

Usage

```
hubers(y, k = 1.5, mu, s, initmu = median(y), tol = 1e-06)
```

Arguments

`y`	vector y of data values
`k`	Winsorizes at k standard deviations
`mu`	specified location
`s`	specified scale
`initmu`	initial value of `mu`
`tol`	convergence tolerance

Value

list of location and scale estimates

`mu`	location estimate
`s`	scale estimate

References

Huber, P. J. (1981) *Robust Statistics.* Wiley.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

See Also

`huber`

Examples

```
hubers(chem)
hubers(chem, mu=3.68)
```

`immer` *Yields from a Barley Field Trial*

Description

The `immer` data frame has 30 rows and 4 columns. Five varieties of barley were grown in six locations in each of 1931 and 1932.

Usage

```
immer
```

Format

This data frame contains the following columns:

`Loc` The location.

`Var` The variety of barley (`"manchuria"`, `"svansota"`, `"velvet"`, `"trebi"` and `"peatland"`).

`Y1` Yield in 1931.

`Y2` Yield in 1932.

Source

Immer, F.R., Hayes, H.D. and LeRoy Powers (1934) Statistical determination of barley varietal adaptation. *Journal of the American Society for Agronomy* **26**, 403–419.

Fisher, R.A. (1947) *The Design of Experiments.* 4th edition. Edinburgh: Oliver and Boyd.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

Examples

```
immer.aov <- aov(cbind(Y1,Y2) ~ Loc + Var, data = immer)
summary(immer.aov)

immer.aov <- aov((Y1+Y2)/2 ~ Var + Loc, data = immer)
summary(immer.aov)
model.tables(immer.aov, type = "means", se = TRUE, cterms = "Var")
```

`Insurance` *Numbers of Car Insurance claims*

Description

The data given in data frame `Insurance` consist of the numbers of policyholders of an insurance company who were exposed to risk, and the numbers of car insurance claims made by those policyholders in the third quarter of 1973.

Usage

```
Insurance
```

Format

This data frame contains the following columns:

`District` factor: district of residence of policyholder (1 to 4): 4 is major cities.

`Group` an ordered factor: group of car with levels <1 litre, 1–1.5 litre, 1.5–2 litre, >2 litre.

`Age` an ordered factor: the age of the insured in 4 groups labelled <25, 25–29, 30–35, >35.

`Holders` numbers of policyholders.

`Claims` numbers of claims

Source

L. A. Baxter, S. M. Coutts and G. A. F. Ross (1980) Applications of linear models in motor insurance. *Proceedings of the 21st International Congress of Actuaries, Zurich* pp. 11–29.

M. Aitkin, D. Anderson, B. Francis and J. Hinde (1989) *Statistical Modelling in GLIM*. Oxford University Press.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS*. Fourth Edition. Springer.

Examples

```
## main-effects fit as Poisson GLM with offset
glm(Claims ~ District + Group + Age + offset(log(Holders)),
    data = Insurance, family = poisson)

# same via loglm
loglm(Claims ~ District + Group + Age + offset(log(Holders)),
      data = Insurance)
```

`isoMDS` *Kruskal's Non-metric Multidimensional Scaling*

Description

One form of non-metric multidimensional scaling

Usage

```
isoMDS(d, y = cmdscale(d, k), k = 2, maxit = 50, trace = TRUE,
       tol = 1e-3, p = 2)

Shepard(d, x, p = 2)
```

Arguments

`d`	distance structure of the form returned by `dist`, or a full, symmetric matrix. Data are assumed to be dissimilarities or relative distances, but must be positive except for self-distance. Both missing and infinite values are allowed.
`y`	An initial configuration. If none is supplied, `cmdscale` is used to provide the classical solution, unless there are missing or infinite dissimilarities.
`k`	The desired dimension for the solution, passed to `cmdscale`.
`maxit`	The maximum number of iterations.
`trace`	Logical for tracing optimization. Default `TRUE`.
`tol`	convergence tolerance.
`p`	Power for Minkowski distance in the configuration space.
`x`	A final configuration.

Details

This chooses a k-dimensional (default k = 2) configuration to minimize the stress, the square root of the ratio of the sum of squared differences between the input distances and those of the configuration to the sum of configuration distances squared. However, the input distances are allowed a monotonic transformation.

An iterative algorithm is used, which will usually converge in around 10 iterations. As this is necessarily an $O(n^2)$ calculation, it is slow for large datasets. Further, since for the default $p = 2$ the configuration is only determined up to rotations and reflections (by convention the centroid is at the origin), the result can vary considerably from machine to machine.

Value

Two components:

`points`	A k-column vector of the fitted configuration.
`stress`	The final stress achieved (in percent).

Side Effects

If `trace` is true, the initial stress and the current stress are printed out every 5 iterations.

References

T. F. Cox and M. A. A. Cox (1994, 2001) *Multidimensional Scaling*. Chapman & Hall.

Ripley, B. D. (1996) *Pattern Recognition and Neural Networks*. Cambridge University Press.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

`cmdscale`, `sammon`

Examples

```
swiss.x <- as.matrix(swiss[, -1])
swiss.dist <- dist(swiss.x)
swiss.mds <- isoMDS(swiss.dist)
plot(swiss.mds$points, type = "n")
text(swiss.mds$points, labels = as.character(1:nrow(swiss.x)))
swiss.sh <- Shepard(swiss.dist, swiss.mds$points)
plot(swiss.sh, pch = ".")
lines(swiss.sh$x, swiss.sh$yf, type = "S")
```

---

kde2d *Two-Dimensional Kernel Density Estimation*

Description

Two-dimensional kernel density estimation with an axis-aligned bivariate normal kernel, evaluated on a square grid.

Usage

```
kde2d(x, y, h, n = 25, lims = c(range(x), range(y)))
```

Arguments

`x`	x coordinate of data
`y`	y coordinate of data
`h`	vector of bandwidths for x and y directions. Defaults to normal reference bandwidth (see `bandwidth.nrd`). A scalar value will be taken to apply to both directions.
`n`	Number of grid points in each direction. Can be scalar or a length-2 integer vector.
`lims`	The limits of the rectangle covered by the grid as `c(xl, xu, yl, yu)`.

Value

A list of three components.

x, y	The x and y coordinates of the grid points, vectors of length n.
z	An n[1] by n[2] matrix of the estimated density: rows correspond to the value of x, columns to the value of y.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

Examples

```
attach(geyser)
plot(duration, waiting, xlim = c(0.5,6), ylim = c(40,100))
f1 <- kde2d(duration, waiting, n = 50, lims = c(0.5, 6, 40, 100))
image(f1, zlim = c(0, 0.05))
f2 <- kde2d(duration, waiting, n = 50, lims = c(0.5, 6, 40, 100),
            h = c(width.SJ(duration), width.SJ(waiting)) )
image(f2, zlim = c(0, 0.05))
persp(f2, phi = 30, theta = 20, d = 5)

plot(duration[-272], duration[-1], xlim = c(0.5, 6),
     ylim = c(1, 6),xlab = "previous duration", ylab = "duration")
f1 <- kde2d(duration[-272], duration[-1],
            h = rep(1.5, 2), n = 50, lims = c(0.5, 6, 0.5, 6))
contour(f1, xlab = "previous duration",
        ylab = "duration", levels  =  c(0.05, 0.1, 0.2, 0.4) )
f1 <- kde2d(duration[-272], duration[-1],
            h = rep(0.6, 2), n = 50, lims = c(0.5, 6, 0.5, 6))
contour(f1, xlab = "previous duration",
        ylab = "duration", levels  =  c(0.05, 0.1, 0.2, 0.4) )
f1 <- kde2d(duration[-272], duration[-1],
            h = rep(0.4, 2), n = 50, lims = c(0.5, 6, 0.5, 6))
contour(f1, xlab = "previous duration",
        ylab = "duration", levels  =  c(0.05, 0.1, 0.2, 0.4) )
detach("geyser")
```

---

lda *Linear Discriminant Analysis*

Description

Linear discriminant analysis.

Usage

```
lda(x, ...)

## S3 method for class 'formula'
lda(formula, data, ..., subset, na.action)

## Default S3 method:
lda(x, grouping, prior = proportions, tol = 1.0e-4,
    method, CV = FALSE, nu, ...)

## S3 method for class 'data.frame'
lda(x, ...)

## S3 method for class 'matrix'
lda(x, grouping, ..., subset, na.action)
```

Arguments

formula	A formula of the form groups ~ x1 + x2 + ... That is, the response is the grouping factor and the right hand side specifies the (non-factor) discriminators.
data	An optional data frame, list or environment from which variables specified in formula are preferentially to be taken.
x	(required if no formula is given as the principal argument.) a matrix or data frame or Matrix containing the explanatory variables.
grouping	(required if no formula principal argument is given.) a factor specifying the class for each observation.
prior	the prior probabilities of class membership. If unspecified, the class proportions for the training set are used. If present, the probabilities should be specified in the order of the factor levels.
tol	A tolerance to decide if a matrix is singular; it will reject variables and linear combinations of unit-variance variables whose variance is less than tol^2.
subset	An index vector specifying the cases to be used in the training sample. (NOTE: If given, this argument must be named.)
na.action	A function to specify the action to be taken if NAs are found. The default action is for the procedure to fail. An alternative is na.omit, which leads to rejection of cases with missing values on any required variable. (NOTE: If given, this argument must be named.)
method	"moment" for standard estimators of the mean and variance, "mle" for MLEs, "mve" to use cov.mve, or "t" for robust estimates based on a t distribution.
CV	If true, returns results (classes and posterior probabilities) for leave-one-out cross-validation. Note that if the prior is estimated, the proportions in the whole dataset are used.
nu	degrees of freedom for method = "t".
...	arguments passed to or from other methods.

Details

The function tries hard to detect if the within-class covariance matrix is singular. If any variable has within-group variance less than `tol^2` it will stop and report the variable as constant. This could result from poor scaling of the problem, but is more likely to result from constant variables.

Specifying the `prior` will affect the classification unless over-ridden in `predict.lda`. Unlike in most statistical packages, it will also affect the rotation of the linear discriminants within their space, as a weighted between-groups covariance matrix is used. Thus the first few linear discriminants emphasize the differences between groups with the weights given by the prior, which may differ from their prevalence in the dataset.

If one or more groups is missing in the supplied data, they are dropped with a warning, but the classifications produced are with respect to the original set of levels.

Value

If `CV = TRUE` the return value is a list with components `class`, the MAP classification (a factor), and `posterior`, posterior probabilities for the classes.

Otherwise it is an object of class `"lda"` containing the following components:

`prior`	the prior probabilities used.
`means`	the group means.
`scaling`	a matrix which transforms observations to discriminant functions, normalized so that within groups covariance matrix is spherical.
`svd`	the singular values, which give the ratio of the between- and within-group standard deviations on the linear discriminant variables. Their squares are the canonical F-statistics.
`N`	The number of observations used.
`call`	The (matched) function call.

Note

This function may be called giving either a formula and optional data frame, or a matrix and grouping factor as the first two arguments. All other arguments are optional, but `subset=` and `na.action=`, if required, must be fully named.

If a formula is given as the principal argument the object may be modified using `update()` in the usual way.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

Ripley, B. D. (1996) *Pattern Recognition and Neural Networks*. Cambridge University Press.

See Also

`predict.lda`, `qda`, `predict.qda`

Examples

```
Iris <- data.frame(rbind(iris3[,,1], iris3[,,2], iris3[,,3]),
                   Sp = rep(c("s","c","v"), rep(50,3)))
train <- sample(1:150, 75)
table(Iris$Sp[train])
## your answer may differ
##  c  s  v
## 22 23 30
z <- lda(Sp ~ ., Iris, prior = c(1,1,1)/3, subset = train)
predict(z, Iris[-train, ])$class
##  [1] s s s s s s s s s s s s s s s s s s s s s s s s s c c c
## [31] c c c c c c c v c c c c v c c c c c c c c c c v v v v v
## [61] v v v v v v v v v v v v v v v v
(z1 <- update(z, . ~ . - Petal.W.))
```

`ldahist` *Histograms or Density Plots of Multiple Groups*

Description

Plot histograms or density plots of data on a single Fisher linear discriminant.

Usage

```
ldahist(data, g, nbins = 25, h, x0 = - h/1000, breaks,
        xlim = range(breaks), ymax = 0, width,
        type = c("histogram", "density", "both"),
        sep = (type != "density"),
        col = 5, xlab = deparse(substitute(data)), bty = "n", ...)
```

Arguments

`data`	vector of data. Missing values (`NA`s) are allowed and omitted.
`g`	factor or vector giving groups, of the same length as `data`.
`nbins`	Suggested number of bins to cover the whole range of the data.
`h`	The bin width (takes precedence over `nbins`).
`x0`	Shift for the bins - the breaks are at `x0 + h * (..., -1, 0, 1, ...)`
`breaks`	The set of breakpoints to be used. (Usually omitted, takes precedence over `h` and `nbins`).
`xlim`	The limits for the x-axis.
`ymax`	The upper limit for the y-axis.
`width`	Bandwidth for density estimates. If missing, the Sheather-Jones selector is used for each group separately.
`type`	Type of plot.
`sep`	Whether there is a separate plot for each group, or one combined plot.

col	The colour number for the bar fill.
xlab	label for the plot x-axis. By default, this will be the name of data.
bty	The box type for the plot - defaults to none.
...	additional arguments to polygon.

Side Effects

Histogram and/or density plots are plotted on the current device.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

plot.lda.

---

leuk *Survival Times and White Blood Counts for Leukaemia Patients*

Description

A data frame of data from 33 leukaemia patients.

Usage

```
leuk
```

Format

A data frame with columns:

wbc white blood count.

ag a test result, "present" or "absent".

time survival time in weeks.

Details

Survival times are given for 33 patients who died from acute myelogenous leukaemia. Also measured was the patient's white blood cell count at the time of diagnosis. The patients were also factored into 2 groups according to the presence or absence of a morphologic characteristic of white blood cells. Patients termed AG positive were identified by the presence of Auer rods and/or significant granulation of the leukaemic cells in the bone marrow at the time of diagnosis.

Source

Cox, D. R. and Oakes, D. (1984) *Analysis of Survival Data*. Chapman & Hall, p. 9.

Taken from

Feigl, P. & Zelen, M. (1965) Estimation of exponential survival probabilities with concomitant information. *Biometrics* **21**, 826–838.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

Examples

```
library(survival)
plot(survfit(Surv(time) ~ ag, data = leuk), lty = 2:3, col = 2:3)

# now Cox models
leuk.cox <- coxph(Surv(time) ~ ag + log(wbc), leuk)
summary(leuk.cox)
```

---

lm.gls *Fit Linear Models by Generalized Least Squares*

Description

Fit linear models by Generalized Least Squares

Usage

```
lm.gls(formula, data, W, subset, na.action, inverse = FALSE,
       method = "qr", model = FALSE, x = FALSE, y = FALSE,
       contrasts = NULL, ...)
```

Arguments

formula	a formula expression as for regression models, of the form `response ~ predictors`. See the documentation of `formula` for other details.
data	an optional data frame, list or environment in which to interpret the variables occurring in `formula`.
W	a weight matrix.
subset	expression saying which subset of the rows of the data should be used in the fit. All observations are included by default.
na.action	a function to filter missing data.
inverse	logical: if true `W` specifies the inverse of the weight matrix: this is appropriate if a variance matrix is used.
method	method to be used by `lm.fit`.

model	should the model frame be returned?
x	should the design matrix be returned?
y	should the response be returned?
contrasts	a list of contrasts to be used for some or all of
...	additional arguments to lm.fit.

Details

The problem is transformed to uncorrelated form and passed to lm.fit.

Value

An object of class "lm.gls", which is similar to an "lm" object. There is no "weights" component, and only a few "lm" methods will work correctly. As from version 7.1-22 the residuals and fitted values refer to the untransformed problem.

See Also

gls, lm, lm.ridge

---

lm.ridge *Ridge Regression*

Description

Fit a linear model by ridge regression.

Usage

```
lm.ridge(formula, data, subset, na.action, lambda = 0, model = FALSE,
         x = FALSE, y = FALSE, contrasts = NULL, ...)
select(obj)
```

Arguments

formula	a formula expression as for regression models, of the form response ~ predictors. See the documentation of formula for other details. offset terms are allowed.
data	an optional data frame, list or environment in which to interpret the variables occurring in formula.
subset	expression saying which subset of the rows of the data should be used in the fit. All observations are included by default.
na.action	a function to filter missing data.
lambda	A scalar or vector of ridge constants.
model	should the model frame be returned? Not implemented.
x	should the design matrix be returned? Not implemented.

y	should the response be returned? Not implemented.
contrasts	a list of contrasts to be used for some or all of factor terms in the formula. See the contrasts.arg of model.matrix.default.
...	additional arguments to lm.fit.
obj	an R object, such as an "lm.ridge" fit.

Details

If an intercept is present in the model, its coefficient is not penalized. (If you want to penalize an intercept, put in your own constant term and remove the intercept.)

Value

A list with components

coef	matrix of coefficients, one row for each value of lambda. Note that these are not on the original scale and are for use by the coef method.
scales	scalings used on the X matrix.
Inter	was intercept included?
lambda	vector of lambda values
ym	mean of y
xm	column means of x matrix
GCV	vector of GCV values
kHKB	HKB estimate of the ridge constant.
kLW	L-W estimate of the ridge constant.

References

Brown, P. J. (1994) *Measurement, Regression and Calibration* Oxford.

See Also

lm

Examples

```
longley # not the same as the S-PLUS dataset
names(longley)[1] <- "y"
lm.ridge(y ~ ., longley)
plot(lm.ridge(y ~ ., longley,
              lambda = seq(0,0.1,0.001)))
select(lm.ridge(y ~ ., longley,
               lambda = seq(0,0.1,0.0001)))
```

`loglm` *Fit Log-Linear Models by Iterative Proportional Scaling*

Description

This function provides a front-end to the standard function, `loglin`, to allow log-linear models to be specified and fitted in a manner similar to that of other fitting functions, such as `glm`.

Usage

```
loglm(formula, data, subset, na.action, ...)
```

Arguments

`formula`	A linear model formula specifying the log-linear model. If the left-hand side is empty, the `data` argument is required and must be a (complete) array of frequencies. In this case the variables on the right-hand side may be the names of the `dimnames` attribute of the frequency array, or may be the positive integers: 1, 2, 3, ... used as alternative names for the 1st, 2nd, 3rd, ... dimension (classifying factor). If the left-hand side is not empty it specifies a vector of frequencies. In this case the data argument, if present, must be a data frame from which the left-hand side vector and the classifying factors on the right-hand side are (preferentially) obtained. The usual abbreviation of a `.` to stand for 'all other variables in the data frame' is allowed. Any non-factors on the right-hand side of the formula are coerced to factor.
`data`	Numeric array or data frame (or list or environment). In the first case it specifies the array of frequencies; in the second it provides the data frame from which the variables occurring in the formula are preferentially obtained in the usual way. This argument may be the result of a call to `xtabs`.
`subset`	Specifies a subset of the rows in the data frame to be used. The default is to take all rows.
`na.action`	Specifies a method for handling missing observations. The default is to fail if missing values are present.
`...`	May supply other arguments to the function `loglm1`.

Details

If the left-hand side of the formula is empty the `data` argument supplies the frequency array and the right-hand side of the formula is used to construct the list of fixed faces as required by `loglin`. Structural zeros may be specified by giving a `start` argument with those entries set to zero, as described in the help information for `loglin`.

If the left-hand side is not empty, all variables on the right-hand side are regarded as classifying factors and an array of frequencies is constructed. If some cells in the complete array are not specified they are treated as structural zeros. The right-hand side of the formula is again used to construct the list of faces on which the observed and fitted totals must agree, as required by `loglin`. Hence terms such as `a:b`, `a*b` and `a/b` are all equivalent.

Value

An object of class "loglm" conveying the results of the fitted log-linear model. Methods exist for the generic functions print, summary, deviance, fitted, coef, resid, anova and update, which perform the expected tasks. Only log-likelihood ratio tests are allowed using anova.

The deviance is simply an alternative name for the log-likelihood ratio statistic for testing the current model within a saturated model, in accordance with standard usage in generalized linear models.

Warning

If structural zeros are present, the calculation of degrees of freedom may not be correct. loglin itself takes no action to allow for structural zeros. loglm deducts one degree of freedom for each structural zero, but cannot make allowance for gains in error degrees of freedom due to loss of dimension in the model space. (This would require checking the rank of the model matrix, but since iterative proportional scaling methods are developed largely to avoid constructing the model matrix explicitly, the computation is at least difficult.)

When structural zeros (or zero fitted values) are present the estimated coefficients will not be available due to infinite estimates. The deviances will normally continue to be correct, though.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

loglm1, loglin

Examples

```
# The data frames  Cars93, minn38 and quine are available
# in the MASS package.

# Case 1: frequencies specified as an array.
sapply(minn38, function(x) length(levels(x)))
## hs phs fol sex f
##  3   4   7   2 0
##minn38a <- array(0, c(3,4,7,2), lapply(minn38[, -5], levels))
##minn38a[data.matrix(minn38[,-5])] <- minn38$f

## or more simply
minn38a <- xtabs(f ~ ., minn38)

fm <- loglm(~ 1 + 2 + 3 + 4, minn38a)  # numerals as names.
deviance(fm)
## [1] 3711.9
fm1 <- update(fm, .~.^2)
fm2 <- update(fm, .~.^3, print = TRUE)
## 5 iterations: deviation 0.075
anova(fm, fm1, fm2)

# Case 1. An array generated with xtabs.
```

```
loglm(~ Type + Origin, xtabs(~ Type + Origin, Cars93))

# Case 2.  Frequencies given as a vector in a data frame
names(quine)
## [1] "Eth"  "Sex"  "Age"  "Lrn"  "Days"
fm <- loglm(Days ~ .^2, quine)
gm <- glm(Days ~ .^2, poisson, quine)  # check glm.
c(deviance(fm), deviance(gm))          # deviances agree
## [1] 1368.7 1368.7
c(fm$df, gm$df)                        # resid df do not!
c(fm$df, gm$df.residual)               # resid df do not!
## [1] 127 128
# The loglm residual degrees of freedom is wrong because of
# a non-detectable redundancy in the model matrix.
```

`logtrans` *Estimate log Transformation Parameter*

Description

Find and optionally plot the marginal (profile) likelihood for alpha for a transformation model of the form `log(y + alpha) ~ x1 + x2 + ...`.

Usage

```
logtrans(object, ...)

## Default S3 method:
logtrans(object, ..., alpha = seq(0.5, 6, by = 0.25) - min(y),
         plotit = TRUE, interp =, xlab = "alpha",
         ylab = "log Likelihood")

## S3 method for class 'formula'
logtrans(object, data, ...)

## S3 method for class 'lm'
logtrans(object, ...)
```

Arguments

`object`	Fitted linear model object, or formula defining the untransformed model that is `y ~ x1 + x2 + ...`. The function is generic.
`...`	If `object` is a formula, this argument may specify a data frame as for `lm`.
`alpha`	Set of values for the transformation parameter, alpha.
`plotit`	Should plotting be done?

interp	Should the marginal log-likelihood be interpolated with a spline approximation? (Default is TRUE if plotting is to be done and the number of real points is less than 100.)
xlab	as for plot.
ylab	as for plot.
data	optional data argument for lm fit.

Value

List with components x (for alpha) and y (for the marginal log-likelihood values).

Side Effects

A plot of the marginal log-likelihood is produced, if requested, together with an approximate mle and 95% confidence interval.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

boxcox

Examples

```
logtrans(Days ~ Age*Sex*Eth*Lrn, data = quine,
         alpha = seq(0.75, 6.5, length.out = 20))
```

---

lqs *Resistant Regression*

Description

Fit a regression to the *good* points in the dataset, thereby achieving a regression estimator with a high breakdown point. lmsreg and ltsreg are compatibility wrappers.

Usage

```
lqs(x, ...)

## S3 method for class 'formula'
lqs(formula, data, ...,
    method = c("lts", "lqs", "lms", "S", "model.frame"),
    subset, na.action, model = TRUE,
    x.ret = FALSE, y.ret = FALSE, contrasts = NULL)
```

```
## Default S3 method:
lqs(x, y, intercept = TRUE, method = c("lts", "lqs", "lms", "S"),
    quantile, control = lqs.control(...), k0 = 1.548, seed, ...)

lmsreg(...)
ltsreg(...)
```

Arguments

formula	a formula of the form y ~ x1 + x2 + ...
data	an optional data frame, list or environemnt from which variables specified in formula are preferentially to be taken.
subset	an index vector specifying the cases to be used in fitting. (NOTE: If given, this argument must be named exactly.)
na.action	function to specify the action to be taken if NAs are found. The default action is for the procedure to fail. Alternatives include na.omit and na.exclude, which lead to omission of cases with missing values on any required variable. (NOTE: If given, this argument must be named exactly.)
model, x.ret, y.ret	logical. If TRUE the model frame, the model matrix and the response are returned, respectively.
contrasts	an optional list. See the contrasts.arg of model.matrix.default.
x	a matrix or data frame containing the explanatory variables.
y	the response: a vector of length the number of rows of x.
intercept	should the model include an intercept?
method	the method to be used. model.frame returns the model frame: for the others see the Details section. Using lmsreg or ltsreg forces "lms" and "lts" respectively.
quantile	the quantile to be used: see Details. This is over-ridden if method = "lms".
control	additional control items: see Details.
k0	the cutoff / tuning constant used for $\chi()$ and $\psi()$ functions when method = "S", currently corresponding to Tukey's 'biweight'.
seed	the seed to be used for random sampling: see .Random.seed. The current value of .Random.seed will be preserved if it is set..
...	arguments to be passed to lqs.default or lqs.control, see control above and Details.

Details

Suppose there are n data points and p regressors, including any intercept.

The first three methods minimize some function of the sorted squared residuals. For methods "lqs" and "lms" is the quantile squared residual, and for "lts" it is the sum of the quantile smallest squared residuals. "lqs" and "lms" differ in the defaults for quantile, which are floor((n+p+1)/2) and floor((n+1)/2) respectively. For "lts" the default is floor(n/2) + floor((p+1)/2).

The "S" estimation method solves for the scale s such that the average of a function chi of the residuals divided by s is equal to a given constant.

The control argument is a list with components

psamp: the size of each sample. Defaults to p.

nsamp: the number of samples or "best" (the default) or "exact" or "sample". If "sample" the number chosen is min(5*p, 3000), taken from Rousseeuw and Hubert (1997). If "best" exhaustive enumeration is done up to 5000 samples; if "exact" exhaustive enumeration will be attempted however many samples are needed.

adjust: should the intercept be optimized for each sample? Defaults to TRUE.

Value

An object of class "lqs". This is a list with components

crit	the value of the criterion for the best solution found, in the case of method == "S" before IWLS refinement.
sing	character. A message about the number of samples which resulted in singular fits.
coefficients	of the fitted linear model
bestone	the indices of those points fitted by the best sample found (prior to adjustment of the intercept, if requested).
fitted.values	the fitted values.
residuals	the residuals.
scale	estimate(s) of the scale of the error. The first is based on the fit criterion. The second (not present for method == "S") is based on the variance of those residuals whose absolute value is less than 2.5 times the initial estimate.

Note

There seems no reason other than historical to use the lms and lqs options. LMS estimation is of low efficiency (converging at rate $n^{-1/3}$) whereas LTS has the same asymptotic efficiency as an M estimator with trimming at the quartiles (Marazzi, 1993, p.201). LQS and LTS have the same maximal breakdown value of (floor((n-p)/2) + 1)/n attained if floor((n+p)/2) <= quantile <= floor((n+p+1)/2). The only drawback mentioned of LTS is greater computation, as a sort was thought to be required (Marazzi, 1993, p.201) but this is not true as a partial sort can be used (and is used in this implementation).

Adjusting the intercept for each trial fit does need the residuals to be sorted, and may be significant extra computation if n is large and p small.

Opinions differ over the choice of psamp. Rousseeuw and Hubert (1997) only consider p; Marazzi (1993) recommends p+1 and suggests that more samples are better than adjustment for a given computational limit.

The computations are exact for a model with just an intercept and adjustment, and for LQS for a model with an intercept plus one regressor and exhaustive search with adjustment. For all other cases the minimization is only known to be approximate.

References

P. J. Rousseeuw and A. M. Leroy (1987) *Robust Regression and Outlier Detection.* Wiley.

A. Marazzi (1993) *Algorithms, Routines and S Functions for Robust Statistics.* Wadsworth and Brooks/Cole.

P. Rousseeuw and M. Hubert (1997) Recent developments in PROGRESS. In *L1-Statistical Procedures and Related Topics*, ed Y. Dodge, IMS Lecture Notes volume **31**, pp. 201–214.

See Also

`predict.lqs`

Examples

```
## IGNORE_RDIFF_BEGIN
set.seed(123) # make reproducible
lqs(stack.loss ~ ., data = stackloss)
lqs(stack.loss ~ ., data = stackloss, method = "S", nsamp = "exact")
## IGNORE_RDIFF_END
```

`mammals` *Brain and Body Weights for 62 Species of Land Mammals*

Description

A data frame with average brain and body weights for 62 species of land mammals.

Usage

```
mammals
```

Format

`body` body weight in kg.

`brain` brain weight in g.

`name` Common name of species. (Rock hyrax-a = *Heterohyrax brucci*, Rock hyrax-b = *Procavia habessinic.*.)

Source

Weisberg, S. (1985) *Applied Linear Regression.* 2nd edition. Wiley, pp. 144–5.

Selected from: Allison, T. and Cicchetti, D. V. (1976) Sleep in mammals: ecological and constitutional correlates. *Science* **194**, 732–734.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

mca *Multiple Correspondence Analysis*

Description

Computes a multiple correspondence analysis of a set of factors.

Usage

```
mca(df, nf = 2, abbrev = FALSE)
```

Arguments

df	A data frame containing only factors
nf	The number of dimensions for the MCA. Rarely 3 might be useful.
abbrev	Should the vertex names be abbreviated? By default these are of the form 'factor.level' but if abbrev = TRUE they are just 'level' which will suffice if the factors have distinct levels.

Value

An object of class "mca", with components

rs	The coordinates of the rows, in nf dimensions.
cs	The coordinates of the column vertices, one for each level of each factor.
fs	Weights for each row, used to interpolate additional factors in predict.mca.
p	The number of factors
d	The singular values for the nf dimensions.
call	The matched call.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

predict.mca, plot.mca, corresp

Examples

```
farms.mca <- mca(farms, abbrev=TRUE)
farms.mca
plot(farms.mca)
```

mcycle *Data from a Simulated Motorcycle Accident*

Description

A data frame giving a series of measurements of head acceleration in a simulated motorcycle accident, used to test crash helmets.

Usage

```
mcycle
```

Format

`times` in milliseconds after impact.

`accel` in g.

Source

Silverman, B. W. (1985) Some aspects of the spline smoothing approach to non-parametric curve fitting. *Journal of the Royal Statistical Society series B* **47**, 1–52.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS*. Fourth Edition. Springer.

Melanoma *Survival from Malignant Melanoma*

Description

The `Melanoma` data frame has data on 205 patients in Denmark with malignant melanoma.

Usage

```
Melanoma
```

Format

This data frame contains the following columns:

`time` survival time in days, possibly censored.

`status` `1` died from melanoma, `2` alive, `3` dead from other causes.

`sex` `1` = male, `0` = female.

`age` age in years.

`year` of operation.

`thickness` tumour thickness in mm.

`ulcer` `1` = presence, `0` = absence.

Source

P. K. Andersen, O. Borgan, R. D. Gill and N. Keiding (1993) *Statistical Models based on Counting Processes.* Springer.

---

menarche *Age of Menarche in Warsaw*

Description

Proportions of female children at various ages during adolescence who have reached menarche.

Usage

```
menarche
```

Format

This data frame contains the following columns:

`Age` Average age of the group. (The groups are reasonably age homogeneous.)

`Total` Total number of children in the group.

`Menarche` Number who have reached menarche.

Source

Milicer, H. and Szczotka, F. (1966) Age at Menarche in Warsaw girls in 1965. *Human Biology* **38**, 199–203.

The data are also given in
Aranda-Ordaz, F.J. (1981) On two families of transformations to additivity for binary response data. *Biometrika* **68**, 357–363.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

Examples

```
mprob <- glm(cbind(Menarche, Total - Menarche) ~ Age,
             binomial(link = probit), data = menarche)
```

`michelson` *Michelson's Speed of Light Data*

Description

Measurements of the speed of light in air, made between 5th June and 2nd July, 1879. The data consists of five experiments, each consisting of 20 consecutive runs. The response is the speed of light in km/s, less 299000. The currently accepted value, on this scale of measurement, is 734.5.

Usage

```
michelson
```

Format

The data frame contains the following components:

`Expt` The experiment number, from 1 to 5.

`Run` The run number within each experiment.

`Speed` Speed-of-light measurement.

Source

A.J. Weekes (1986) *A Genstat Primer.* Edward Arnold.

S. M. Stigler (1977) Do robust estimators work with real data? *Annals of Statistics* **5**, 1055–1098.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

minn38 *Minnesota High School Graduates of 1938*

Description

The Minnesota high school graduates of 1938 were classified according to four factors, described below. The `minn38` data frame has 168 rows and 5 columns.

Usage

```
minn38
```

Format

This data frame contains the following columns:

`hs` high school rank: `"L"`, `"M"` and `"U"` for lower, middle and upper third.

`phs` post high school status: Enrolled in college, (`"C"`), enrolled in non-collegiate school, (`"N"`), employed full-time, (`"E"`) and other, (`"O"`).

`fol` father's occupational level, (seven levels, `"F1"`, `"F2"`, ..., `"F7"`).

`sex` sex: factor with levels`"F"` or `"M"`.

`f` frequency.

Source

From R. L. Plackett, (1974) *The Analysis of Categorical Data.* London: Griffin

who quotes the data from

Hoyt, C. J., Krishnaiah, P. R. and Torrance, E. P. (1959) Analysis of complex contingency tables, *J. Exp. Ed.* **27**, 187–194.

motors *Accelerated Life Testing of Motorettes*

Description

The `motors` data frame has 40 rows and 3 columns. It describes an accelerated life test at each of four temperatures of 10 motorettes, and has rather discrete times.

Usage

```
motors
```

Format

This data frame contains the following columns:

`temp` the temperature (degrees C) of the test.

`time` the time in hours to failure or censoring at 8064 hours (= 336 days).

`cens` an indicator variable for death.

Source

Kalbfleisch, J. D. and Prentice, R. L. (1980) *The Statistical Analysis of Failure Time Data.* New York: Wiley.

taken from

Nelson, W. D. and Hahn, G. J. (1972) Linear regression of a regression relationship from censored data. Part 1 – simple methods and their application. *Technometrics*, **14**, 247–276.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

Examples

```
library(survival)
plot(survfit(Surv(time, cens) ~ factor(temp), motors), conf.int = FALSE)
# fit Weibull model
motor.wei <- survreg(Surv(time, cens) ~ temp, motors)
summary(motor.wei)
# and predict at 130C
unlist(predict(motor.wei, data.frame(temp=130), se.fit = TRUE))

motor.cox <- coxph(Surv(time, cens) ~ temp, motors)
summary(motor.cox)
# predict at temperature 200
plot(survfit(motor.cox, newdata = data.frame(temp=200),
     conf.type = "log-log"))
summary( survfit(motor.cox, newdata = data.frame(temp=130)) )
```

---

muscle *Effect of Calcium Chloride on Muscle Contraction in Rat Hearts*

Description

The purpose of this experiment was to assess the influence of calcium in solution on the contraction of heart muscle in rats. The left auricle of 21 rat hearts was isolated and on several occasions a constant-length strip of tissue was electrically stimulated and dipped into various concentrations of calcium chloride solution, after which the shortening of the strip was accurately measured as the response.

Usage

```
muscle
```

Format

This data frame contains the following columns:

`Strip` which heart muscle strip was used?

`Conc` concentration of calcium chloride solution, in multiples of 2.2 mM.

`Length` the change in length (shortening) of the strip, (allegedly) in mm.

Source

Linder, A., Chakravarti, I. M. and Vuagnat, P. (1964) Fitting asymptotic regression curves with different asymptotes. In *Contributions to Statistics. Presented to Professor P. C. Mahalanobis on the occasion of his 70th birthday*, ed. C. R. Rao, pp. 221–228. Oxford: Pergamon Press.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth Edition. Springer.

Examples

```
## IGNORE_RDIFF_BEGIN
A <- model.matrix(~ Strip - 1, data=muscle)
rats.nls1 <- nls(log(Length) ~ cbind(A, rho^Conc),
   data = muscle, start = c(rho=0.1), algorithm="plinear")
(B <- coef(rats.nls1))

st <- list(alpha = B[2:22], beta = B[23], rho = B[1])
(rats.nls2 <- nls(log(Length) ~ alpha[Strip] + beta*rho^Conc,
                  data = muscle, start = st))
## IGNORE_RDIFF_END

Muscle <- with(muscle, {
Muscle <- expand.grid(Conc = sort(unique(Conc)), Strip = levels(Strip))
Muscle$Yhat <- predict(rats.nls2, Muscle)
Muscle <- cbind(Muscle, logLength = rep(as.numeric(NA), 126))
ind <- match(paste(Strip, Conc),
            paste(Muscle$Strip, Muscle$Conc))
Muscle$logLength[ind] <- log(Length)
Muscle})

lattice::xyplot(Yhat ~ Conc | Strip, Muscle, as.table = TRUE,
   ylim = range(c(Muscle$Yhat, Muscle$logLength), na.rm = TRUE),
   subscripts = TRUE, xlab = "Calcium Chloride concentration (mM)",
   ylab = "log(Length in mm)", panel =
   function(x, y, subscripts, ...) {
      panel.xyplot(x, Muscle$logLength[subscripts], ...)
      llines(spline(x, y))
   })
```

mvrnorm *Simulate from a Multivariate Normal Distribution*

Description

Produces one or more samples from the specified multivariate normal distribution.

Usage

```
mvrnorm(n = 1, mu, Sigma, tol = 1e-6, empirical = FALSE, EISPACK = FALSE)
```

Arguments

n	the number of samples required.
mu	a vector giving the means of the variables.
Sigma	a positive-definite symmetric matrix specifying the covariance matrix of the variables.
tol	tolerance (relative to largest variance) for numerical lack of positive-definiteness in Sigma.
empirical	logical. If true, mu and Sigma specify the empirical not population mean and covariance matrix.
EISPACK	logical: values other than FALSE are an error.

Details

The matrix decomposition is done via eigen; although a Choleski decomposition might be faster, the eigendecomposition is stabler.

Value

If n = 1 a vector of the same length as mu, otherwise an n by length(mu) matrix with one sample in each row.

Side Effects

Causes creation of the dataset .Random.seed if it does not already exist, otherwise its value is updated.

References

B. D. Ripley (1987) *Stochastic Simulation.* Wiley. Page 98.

See Also

rnorm

Examples

```
Sigma <- matrix(c(10,3,3,2),2,2)
Sigma
var(mvrnorm(n = 1000, rep(0, 2), Sigma))
var(mvrnorm(n = 1000, rep(0, 2), Sigma, empirical = TRUE))
```

`negative.binomial` *Family function for Negative Binomial GLMs*

Description

Specifies the information required to fit a Negative Binomial generalized linear model, with known `theta` parameter, using `glm()`.

Usage

```
negative.binomial(theta = stop("'theta' must be specified"), link = "log")
```

Arguments

`theta`	The known value of the additional parameter, `theta`.
`link`	The link function, as a character string, name or one-element character vector specifying one of `log`, `sqrt` or `identity`, or an object of class `"link-glm"`.

Value

An object of class `"family"`, a list of functions and expressions needed by `glm()` to fit a Negative Binomial generalized linear model.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

See Also

`glm.nb`, `anova.negbin`, `summary.negbin`

Examples

```
# Fitting a Negative Binomial model to the quine data
#   with theta = 2 assumed known.
#
glm(Days ~ .^4, family = negative.binomial(2), data = quine)
```

newcomb *Newcomb's Measurements of the Passage Time of Light*

Description

A numeric vector giving the 'Third Series' of measurements of the passage time of light recorded by Newcomb in 1882. The given values divided by 1000 plus 24.8 give the time in millionths of a second for light to traverse a known distance. The 'true' value is now considered to be 33.02.

The dataset is given in the order in Staudte and Sheather. Stigler (1977, Table 5) gives the dataset as

```
28 26 33 24 34 -44 27 16 40 -2 29 22 24 21 25 30 23 29 31 19
24 20 36 32 36 28 25 21 28 29 37 25 28 26 30 32 36 26 30 22
36 23 27 27 28 27 31 27 26 33 26 32 32 24 39 28 24 25 32 25
29 27 28 29 16 23
```

However, order is not relevant to its use as an example of robust estimation. (Thanks to Anthony Unwin for bringing this difference to our attention.)

Usage

```
newcomb
```

Source

S. M. Stigler (1973) Simon Newcomb, Percy Daniell, and the history of robust estimation 1885–1920. *Journal of the American Statistical Association* **68**, 872–879.

S. M. Stigler (1977) Do robust estimators work with *real* data? *Annals of Statistics*, **5**, 1055–1098.

R. G. Staudte and S. J. Sheather (1990) *Robust Estimation and Testing.* Wiley.

nlschools *Eighth-Grade Pupils in the Netherlands*

Description

Snijders and Bosker (1999) use as a running example a study of 2287 eighth-grade pupils (aged about 11) in 132 classes in 131 schools in the Netherlands. Only the variables used in our examples are supplied.

Usage

```
nlschools
```

Format

This data frame contains 2287 rows and the following columns:

`lang` language test score.

`IQ` verbal IQ.

`class` class ID.

`GS` class size: number of eighth-grade pupils recorded in the class (there may be others: see `COMB`, and some may have been omitted with missing values).

`SES` social-economic status of pupil's family.

`COMB` were the pupils taught in a multi-grade class (`0/1`)? Classes which contained pupils from grades 7 and 8 are coded 1, but only eighth-graders were tested.

Source

Snijders, T. A. B. and Bosker, R. J. (1999) *Multilevel Analysis. An Introduction to Basic and Advanced Multilevel Modelling.* London: Sage.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

Examples

```
nl1 <- within(nlschools, {
IQave <- tapply(IQ, class, mean)[as.character(class)]
IQ <- IQ - IQave
})
cen <- c("IQ", "IQave", "SES")
nl1[cen] <- scale(nl1[cen], center = TRUE, scale = FALSE)

nl.lme <- nlme::lme(lang ~ IQ*COMB + IQave + SES,
                    random = ~ IQ | class, data = nl1)
## IGNORE_RDIFF_BEGIN
summary(nl.lme)
## IGNORE_RDIFF_END
```

---

npk *Classical N, P, K Factorial Experiment*

Description

A classical N, P, K (nitrogen, phosphate, potassium) factorial experiment on the growth of peas conducted on 6 blocks. Each half of a fractional factorial design confounding the NPK interaction was used on 3 of the plots.

Usage

```
npk
```

Format

The npk data frame has 24 rows and 5 columns:

`block` which block (label 1 to 6).

`N` indicator (0/1) for the application of nitrogen.

`P` indicator (0/1) for the application of phosphate.

`K` indicator (0/1) for the application of potassium.

`yield` Yield of peas, in pounds/plot (the plots were (1/70) acre).

Note

This dataset is also contained in R 3.0.2 and later.

Source

Imperial College, London, M.Sc. exercise sheet.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

Examples

```
options(contrasts = c("contr.sum", "contr.poly"))
npk.aov <- aov(yield ~ block + N*P*K, npk)
## IGNORE_RDIFF_BEGIN
npk.aov
summary(npk.aov)
alias(npk.aov)
coef(npk.aov)
options(contrasts = c("contr.treatment", "contr.poly"))
npk.aov1 <- aov(yield ~ block + N + K, data = npk)
summary.lm(npk.aov1)
se.contrast(npk.aov1, list(N=="0", N=="1"), data = npk)
model.tables(npk.aov1, type = "means", se = TRUE)
## IGNORE_RDIFF_END
```

npr1 *US Naval Petroleum Reserve No. 1 data*

Description

Data on the locations, porosity and permeability (a measure of oil flow) on 104 oil wells in the US Naval Petroleum Reserve No. 1 in California.

Usage

```
npr1
```

Format

This data frame contains the following columns:

`x` x coordinates, in miles (origin unspecified)..

`y` y coordinates, in miles.

`perm` permeability in milli-Darcies.

`por` porosity (%).

Source

Maher, J.C., Carter, R.D. and Lantz, R.J. (1975) Petroleum geology of Naval Petroleum Reserve No. 1, Elk Hills, Kern County, California. *USGS Professional Paper* **912**.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

Null *Null Spaces of Matrices*

Description

Given a matrix, `M`, find a matrix `N` giving a basis for the (left) null space. That is `crossprod(N, M) = t(N) %*% M` is an all-zero matrix and `N` has the maximum number of linearly independent columns.

Usage

```
Null(M)
```

Arguments

`M`	Input matrix. A vector is coerced to a 1-column matrix.

Details

For a basis for the (right) null space $\{x : Mx = 0\}$, use `Null(t(M))`.

Value

The matrix `N` with the basis for the (left) null space, or a matrix with zero columns if the matrix `M` is square and of maximal rank.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

`qr`, `qr.Q`.

Examples

```
# The function is currently defined as
function(M)
{
    tmp <- qr(M)
    set <- if(tmp$rank == 0L) seq_len(ncol(M)) else  -seq_len(tmp$rank)
    qr.Q(tmp, complete = TRUE)[, set, drop = FALSE]
}
```

---

oats *Data from an Oats Field Trial*

Description

The yield of oats from a split-plot field trial using three varieties and four levels of manurial treatment. The experiment was laid out in 6 blocks of 3 main plots, each split into 4 sub-plots. The varieties were applied to the main plots and the manurial treatments to the sub-plots.

Usage

```
oats
```

Format

This data frame contains the following columns:

`B` Blocks, levels I, II, III, IV, V and VI.

`V` Varieties, 3 levels.

`N` Nitrogen (manurial) treatment, levels 0.0cwt, 0.2cwt, 0.4cwt and 0.6cwt, showing the application in cwt/acre.

`Y` Yields in 1/4lbs per sub-plot, each of area 1/80 acre.

Source

Yates, F. (1935) Complex experiments, *Journal of the Royal Statistical Society Suppl.* **2**, 181–247.

Also given in Yates, F. (1970) *Experimental design: Selected papers of Frank Yates, C.B.E, F.R.S.* London: Griffin.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

Examples

```
oats$Nf <- ordered(oats$N, levels = sort(levels(oats$N)))
oats.aov <- aov(Y ~ Nf*V + Error(B/V), data = oats, qr = TRUE)
## IGNORE_RDIFF_BEGIN
summary(oats.aov)
summary(oats.aov, split = list(Nf=list(L=1, Dev=2:3)))
## IGNORE_RDIFF_END
par(mfrow = c(1,2), pty = "s")
plot(fitted(oats.aov[[4]]), studres(oats.aov[[4]]))
abline(h = 0, lty = 2)
oats.pr <- proj(oats.aov)
qqnorm(oats.pr[[4]][,"Residuals"], ylab = "Stratum 4 residuals")
qqline(oats.pr[[4]][,"Residuals"])

par(mfrow = c(1,1), pty = "m")
oats.aov2 <- aov(Y ~ N + V + Error(B/V), data = oats, qr = TRUE)
model.tables(oats.aov2, type = "means", se = TRUE)
```

OME *Tests of Auditory Perception in Children with OME*

Description

Experiments were performed on children on their ability to differentiate a signal in broad-band noise. The noise was played from a pair of speakers and a signal was added to just one channel; the subject had to turn his/her head to the channel with the added signal. The signal was either coherent (the amplitude of the noise was increased for a period) or incoherent (independent noise was added for the same period to form the same increase in power).

The threshold used in the original analysis was the stimulus loudness needs to get 75% correct responses. Some of the children had suffered from otitis media with effusion (OME).

Usage

```
OME
```

Format

The OME data frame has 1129 rows and 7 columns:

ID Subject ID (1 to 99, with some IDs missing). A few subjects were measured at different ages.

OME "low" or "high" or "N/A" (at ages other than 30 and 60 months).

Age Age of the subject (months).

Loud Loudness of stimulus, in decibels.

Noise Whether the signal in the stimulus was "coherent" or "incoherent".

Correct Number of correct responses from Trials trials.

Trials Number of trials performed.

Background

The experiment was to study otitis media with effusion (OME), a very common childhood condition where the middle ear space, which is normally air-filled, becomes congested by a fluid. There is a concomitant fluctuating, conductive hearing loss which can result in various language, cognitive and social deficits. The term 'binaural hearing' is used to describe the listening conditions in which the brain is processing information from both ears at the same time. The brain computes differences in the intensity and/or timing of signals arriving at each ear which contributes to sound localisation and also to our ability to hear in background noise.

Some years ago, it was found that children of 7–8 years with a history of significant OME had significantly worse binaural hearing than children without such a history, despite having equivalent sensitivity. The question remained as to whether it was the timing, the duration, or the degree of severity of the otitis media episodes during critical periods, which affected later binaural hearing. In an attempt to begin to answer this question, 95 children were monitored for the presence of effusion every month since birth. On the basis of OME experience in their first two years, the test population was split into one group of high OME prevalence and one of low prevalence.

Source

Sarah Hogan, Dept of Physiology, University of Oxford, via Dept of Statistics Consulting Service

Examples

```
# Fit logistic curve from p = 0.5 to p = 1.0
fp1 <- deriv(~ 0.5 + 0.5/(1 + exp(-(x-L75)/scal)),
             c("L75", "scal"),
             function(x,L75,scal)NULL)
nls(Correct/Trials ~ fp1(Loud, L75, scal), data = OME,
    start = c(L75=45, scal=3))
nls(Correct/Trials ~ fp1(Loud, L75, scal),
    data = OME[OME$Noise == "coherent",],
    start=c(L75=45, scal=3))
nls(Correct/Trials ~ fp1(Loud, L75, scal),
    data = OME[OME$Noise == "incoherent",],
    start = c(L75=45, scal=3))

# individual fits for each experiment
```

```
aa <- factor(OME$Age)
ab <- 10*OME$ID + unclass(aa)
ac <- unclass(factor(ab))
OME$UID <- as.vector(ac)
OME$UIDn <- OME$UID + 0.1*(OME$Noise == "incoherent")
rm(aa, ab, ac)
OMEi <- OME

library(nlme)
fp2 <- deriv(~ 0.5 + 0.5/(1 + exp(-(x-L75)/2)),
           "L75", function(x,L75) NULL)
dec <- getOption("OutDec")
options(show.error.messages = FALSE, OutDec=".")
OMEi.nls <- nlsList(Correct/Trials ~ fp2(Loud, L75) | UIDn,
   data = OMEi, start = list(L75=45), control = list(maxiter=100))
options(show.error.messages = TRUE, OutDec=dec)
tmp <- sapply(OMEi.nls, function(X)
             {if(is.null(X)) NA else as.vector(coef(X))})
OMEif <- data.frame(UID = round(as.numeric((names(tmp)))),
         Noise = rep(c("coherent", "incoherent"), 110),
         L75 = as.vector(tmp), stringsAsFactors = TRUE)
OMEif$Age <- OME$Age[match(OMEif$UID, OME$UID)]
OMEif$OME <- OME$OME[match(OMEif$UID, OME$UID)]
OMEif <- OMEif[OMEif$L75 > 30,]
summary(lm(L75 ~ Noise/Age, data = OMEif, na.action = na.omit))
summary(lm(L75 ~ Noise/(Age + OME), data = OMEif,
           subset = (Age >= 30 & Age <= 60),
           na.action = na.omit), correlation = FALSE)

# Or fit by weighted least squares
fpl75 <- deriv(~ sqrt(n)*(r/n - 0.5 - 0.5/(1 + exp(-(x-L75)/scal))),
               c("L75", "scal"),
               function(r,n,x,L75,scal) NULL)
nls(0 ~ fpl75(Correct, Trials, Loud, L75, scal),
    data = OME[OME$Noise == "coherent",],
    start = c(L75=45, scal=3))
nls(0 ~ fpl75(Correct, Trials, Loud, L75, scal),
    data = OME[OME$Noise == "incoherent",],
    start = c(L75=45, scal=3))

# Test to see if the curves shift with age
fpl75age <- deriv(~sqrt(n)*(r/n -  0.5 - 0.5/(1 +
                 exp(-(x-L75-slope*age)/scal))),
                 c("L75", "slope", "scal"),
                 function(r,n,x,age,L75,slope,scal) NULL)
OME.nls1 <-
nls(0 ~ fpl75age(Correct, Trials, Loud, Age, L75, slope, scal),
    data = OME[OME$Noise == "coherent",],
    start = c(L75=45, slope=0, scal=2))
sqrt(diag(vcov(OME.nls1)))

OME.nls2 <-
```

```
nls(0 ~ fpl75age(Correct, Trials, Loud, Age, L75, slope, scal),
    data = OME[OME$Noise == "incoherent",],
    start = c(L75=45, slope=0, scal=2))
sqrt(diag(vcov(OME.nls2)))

# Now allow random effects by using NLME
OMEf <- OME[rep(1:nrow(OME), OME$Trials),]
OMEf$Resp <- with(OME, rep(rep(c(1,0), length(Trials)),
                           t(cbind(Correct, Trials-Correct))))
OMEf <- OMEf[, -match(c("Correct", "Trials"), names(OMEf))]

## Not run: ## these fail in R on most platforms
fp2 <- deriv(~ 0.5 + 0.5/(1 + exp(-(x-L75)/exp(lsc))),
             c("L75", "lsc"),
             function(x, L75, lsc) NULL)
try(summary(nlme(Resp ~ fp2(Loud, L75, lsc),
     fixed = list(L75 ~ Age, lsc ~ 1),
     random = L75 + lsc ~ 1 | UID,
     data = OMEf[OMEf$Noise == "coherent",], method = "ML",
     start = list(fixed=c(L75=c(48.7, -0.03), lsc=0.24)), verbose = TRUE)))

try(summary(nlme(Resp ~ fp2(Loud, L75, lsc),
     fixed = list(L75 ~ Age, lsc ~ 1),
     random = L75 + lsc ~ 1 | UID,
     data = OMEf[OMEf$Noise == "incoherent",], method = "ML",
     start = list(fixed=c(L75=c(41.5, -0.1), lsc=0)), verbose = TRUE)))

## End(Not run)
```

`painters` *The Painter's Data of de Piles*

Description

The subjective assessment, on a 0 to 20 integer scale, of 54 classical painters. The painters were assessed on four characteristics: composition, drawing, colour and expression. The data is due to the Eighteenth century art critic, de Piles.

Usage

```
painters
```

Format

The row names of the data frame are the painters. The components are:

`Composition` Composition score.

`Drawing` Drawing score.

`Colour` Colour score.

`Expression` Expression score.

`School` The school to which a painter belongs, as indicated by a factor level code as follows: `"A"`: Renaissance; `"B"`: Mannerist; `"C"`: Seicento; `"D"`: Venetian; `"E"`: Lombard; `"F"`: Sixteenth Century; `"G"`: Seventeenth Century; `"H"`: French.

Source

A. J. Weekes (1986) *A Genstat Primer.* Edward Arnold.

M. Davenport and G. Studdert-Kennedy (1972) The statistical analysis of aesthetic judgement: an exploration. *Applied Statistics* **21**, 324–333.

I. T. Jolliffe (1986) *Principal Component Analysis.* Springer.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

`pairs.lda` *Produce Pairwise Scatterplots from an 'lda' Fit*

Description

Pairwise scatterplot of the data on the linear discriminants.

Usage

```
## S3 method for class 'lda'
pairs(x, labels = colnames(x), panel = panel.lda,
     dimen, abbrev = FALSE, ..., cex=0.7, type = c("std", "trellis"))
```

Arguments

`x`	Object of class `"lda"`.
`labels`	vector of character strings for labelling the variables.
`panel`	panel function to plot the data in each panel.
`dimen`	The number of linear discriminants to be used for the plot; if this exceeds the number determined by `x` the smaller value is used.
`abbrev`	whether the group labels are abbreviated on the plots. If `abbrev > 0` this gives `minlength` in the call to `abbreviate`.
`...`	additional arguments for `pairs.default`.
`cex`	graphics parameter `cex` for labels on plots.
`type`	type of plot. The default is in the style of `pairs.default`; the style `"trellis"` uses the Trellis function `splom`.

Details

This function is a method for the generic function `pairs()` for class `"lda"`. It can be invoked by calling `pairs(x)` for an object `x` of the appropriate class, or directly by calling `pairs.lda(x)` regardless of the class of the object.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

`pairs`

---

parcoord *Parallel Coordinates Plot*

Description

Parallel coordinates plot

Usage

```
parcoord(x, col = 1, lty = 1, var.label = FALSE, ...)
```

Arguments

`x`	a matrix or data frame who columns represent variables. Missing values are allowed.
`col`	A vector of colours, recycled as necessary for each observation.
`lty`	A vector of line types, recycled as necessary for each observation.
`var.label`	If `TRUE`, each variable's axis is labelled with maximum and minimum values.
`...`	Further graphics parameters which are passed to `matplot`.

Side Effects

a parallel coordinates plots is drawn.

Author(s)

B. D. Ripley. Enhancements based on ideas and code by Fabian Scheipl.

References

Wegman, E. J. (1990) Hyperdimensional data analysis using parallel coordinates. *Journal of the American Statistical Association* **85**, 664–675.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

Examples

```
parcoord(state.x77[, c(7, 4, 6, 2, 5, 3)])

ir <- rbind(iris3[,,1], iris3[,,2], iris3[,,3])
parcoord(log(ir)[, c(3, 4, 2, 1)], col = 1 + (0:149)%/%50)
```

`petrol` *N. L. Prater's Petrol Refinery Data*

Description

The yield of a petroleum refining process with four covariates. The crude oil appears to come from only 10 distinct samples.

These data were originally used by Prater (1956) to build an estimation equation for the yield of the refining process of crude oil to gasoline.

Usage

```
petrol
```

Format

The variables are as follows

`No` crude oil sample identification label. (Factor.)

`SG` specific gravity, degrees API. (Constant within sample.)

`VP` vapour pressure in pounds per square inch. (Constant within sample.)

`V10` volatility of crude; ASTM 10% point. (Constant within sample.)

`EP` desired volatility of gasoline. (The end point. Varies within sample.)

`Y` yield as a percentage of crude.

Source

N. H. Prater (1956) Estimate gasoline yields from crudes. *Petroleum Refiner* **35**, 236–238.

This dataset is also given in D. J. Hand, F. Daly, K. McConway, D. Lunn and E. Ostrowski (eds) (1994) *A Handbook of Small Data Sets.* Chapman & Hall.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

Examples

```
library(nlme)
Petrol <- petrol
Petrol[, 2:5] <- scale(as.matrix(Petrol[, 2:5]), scale = FALSE)
pet3.lme <- lme(Y ~ SG + VP + V10 + EP,
                random = ~ 1 | No, data = Petrol)
pet3.lme <- update(pet3.lme, method = "ML")
pet4.lme <- update(pet3.lme, fixed. = Y ~ V10 + EP)
anova(pet4.lme, pet3.lme)
```

Pima.tr *Diabetes in Pima Indian Women*

Description

A population of women who were at least 21 years old, of Pima Indian heritage and living near Phoenix, Arizona, was tested for diabetes according to World Health Organization criteria. The data were collected by the US National Institute of Diabetes and Digestive and Kidney Diseases. We used the 532 complete records after dropping the (mainly missing) data on serum insulin.

Usage

```
Pima.tr
Pima.tr2
Pima.te
```

Format

These data frames contains the following columns:

`npreg` number of pregnancies.

`glu` plasma glucose concentration in an oral glucose tolerance test.

`bp` diastolic blood pressure (mm Hg).

`skin` triceps skin fold thickness (mm).

`bmi` body mass index (weight in kg/(height in m)^2).

`ped` diabetes pedigree function.

`age` age in years.

`type` `Yes` or `No`, for diabetic according to WHO criteria.

Details

The training set `Pima.tr` contains a randomly selected set of 200 subjects, and `Pima.te` contains the remaining 332 subjects. `Pima.tr2` contains `Pima.tr` plus 100 subjects with missing values in the explanatory variables.

Source

Smith, J. W., Everhart, J. E., Dickson, W. C., Knowler, W. C. and Johannes, R. S. (1988) Using the ADAP learning algorithm to forecast the onset of *diabetes mellitus*. In *Proceedings of the Symposium on Computer Applications in Medical Care (Washington, 1988),* ed. R. A. Greenes, pp. 261–265. Los Alamitos, CA: IEEE Computer Society Press.

Ripley, B.D. (1996) *Pattern Recognition and Neural Networks.* Cambridge: Cambridge University Press.

`plot.lda` *Plot Method for Class 'lda'*

Description

Plots a set of data on one, two or more linear discriminants.

Usage

```
## S3 method for class 'lda'
plot(x, panel = panel.lda, ..., cex = 0.7, dimen,
     abbrev = FALSE, xlab = "LD1", ylab = "LD2")
```

Arguments

`x`	An object of class `"lda"`.
`panel`	the panel function used to plot the data.
`...`	additional arguments to `pairs`, `ldahist` or `eqscplot`.
`cex`	graphics parameter `cex` for labels on plots.
`dimen`	The number of linear discriminants to be used for the plot; if this exceeds the number determined by `x` the smaller value is used.
`abbrev`	whether the group labels are abbreviated on the plots. If `abbrev > 0` this gives `minlength` in the call to `abbreviate`.
`xlab`	label for the x axis
`ylab`	label for the y axis

Details

This function is a method for the generic function `plot()` for class `"lda"`. It can be invoked by calling `plot(x)` for an object `x` of the appropriate class, or directly by calling `plot.lda(x)` regardless of the class of the object.

The behaviour is determined by the value of `dimen`. For `dimen > 2`, a `pairs` plot is used. For `dimen = 2`, an equiscaled scatter plot is drawn. For `dimen = 1`, a set of histograms or density plots are drawn. Use argument `type` to match `"histogram"` or `"density"` or `"both"`.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

pairs.lda, ldahist, lda, predict.lda

---

plot.mca *Plot Method for Objects of Class 'mca'*

Description

Plot a multiple correspondence analysis.

Usage

```
## S3 method for class 'mca'
plot(x, rows = TRUE, col, cex = par("cex"), ...)
```

Arguments

x	An object of class "mca".
rows	Should the coordinates for the rows be plotted, or just the vertices for the levels?
col, cex	The colours and cex to be used for the row points and level vertices respectively.
...	Additional parameters to plot.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

mca, predict.mca

Examples

```
plot(mca(farms, abbrev = TRUE))
```

polr *Ordered Logistic or Probit Regression*

Description

Fits a logistic or probit regression model to an ordered factor response. The default logistic case is *proportional odds logistic regression*, after which the function is named.

Usage

```
polr(formula, data, weights, start, ..., subset, na.action,
     contrasts = NULL, Hess = FALSE, model = TRUE,
     method = c("logistic", "probit", "loglog", "cloglog", "cauchit"))
```

Arguments

formula	a formula expression as for regression models, of the form `response ~ predictors`. The response should be a factor (preferably an ordered factor), which will be interpreted as an ordinal response, with levels ordered as in the factor. The model must have an intercept: attempts to remove one will lead to a warning and be ignored. An offset may be used. See the documentation of `formula` for other details.
data	an optional data frame, list or environment in which to interpret the variables occurring in `formula`.
weights	optional case weights in fitting. Default to 1.
start	initial values for the parameters. This is in the format `c(coefficients, zeta)`: see the Values section.
...	additional arguments to be passed to `optim`, most often a `control` argument.
subset	expression saying which subset of the rows of the data should be used in the fit. All observations are included by default.
na.action	a function to filter missing data.
contrasts	a list of contrasts to be used for some or all of the factors appearing as variables in the model formula.
Hess	logical for whether the Hessian (the observed information matrix) should be returned. Use this if you intend to call `summary` or `vcov` on the fit.
model	logical for whether the model matrix should be returned.
method	logistic or probit or (complementary) log-log or cauchit (corresponding to a Cauchy latent variable).

Details

This model is what Agresti (2002) calls a *cumulative link* model. The basic interpretation is as a *coarsened* version of a latent variable Y_i which has a logistic or normal or extreme-value or Cauchy distribution with scale parameter one and a linear model for the mean. The ordered factor which is observed is which bin Y_i falls into with breakpoints

$$\zeta_0 = -\infty < \zeta_1 < \cdots < \zeta_K = \infty$$

This leads to the model

$$\text{logit} P(Y \le k|x) = \zeta_k - \eta$$

with *logit* replaced by *probit* for a normal latent variable, and η being the linear predictor, a linear function of the explanatory variables (with no intercept). Note that it is quite common for other software to use the opposite sign for η (and hence the coefficients `beta`).

In the logistic case, the left-hand side of the last display is the log odds of category k or less, and since these are log odds which differ only by a constant for different k, the odds are proportional. Hence the term *proportional odds logistic regression*.

The log-log and complementary log-log links are the increasing functions $F^{-1}(p) = -log(-log(p))$ and $F^{-1}(p) = log(-log(1-p))$; some call the first the 'negative log-log' link. These correspond to a latent variable with the extreme-value distribution for the maximum and minimum respectively.

A *proportional hazards* model for grouped survival times can be obtained by using the complementary log-log link with grouping ordered by increasing times.

`predict`, `summary`, `vcov`, `anova`, `model.frame` and an `extractAIC` method for use with `stepAIC` (and `step`). There are also `profile` and `confint` methods.

Value

A object of class `"polr"`. This has components

`coefficients`	the coefficients of the linear predictor, which has no intercept.
`zeta`	the intercepts for the class boundaries.
`deviance`	the residual deviance.
`fitted.values`	a matrix, with a column for each level of the response.
`lev`	the names of the response levels.
`terms`	the `terms` structure describing the model.
`df.residual`	the number of residual degrees of freedoms, calculated using the weights.
`edf`	the (effective) number of degrees of freedom used by the model
`n, nobs`	the (effective) number of observations, calculated using the weights. (`nobs` is for use by `stepAIC`.
`call`	the matched call.
`method`	the matched method used.
`convergence`	the convergence code returned by `optim`.
`niter`	the number of function and gradient evaluations used by `optim`.
`lp`	the linear predictor (including any offset).
`Hessian`	(if `Hess` is true). Note that this is a numerical approximation derived from the optimization proces.
`model`	(if `model` is true).

Note

The vcov method uses the approximate Hessian: for reliable results the model matrix should be sensibly scaled with all columns having range the order of one.

Prior to version 7.3-32, `method = "cloglog"` confusingly gave the log-log link, implicitly assuming the first response level was the 'best'.

References

Agresti, A. (2002) *Categorical Data.* Second edition. Wiley.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

See Also

optim, glm, multinom.

Examples

```
options(contrasts = c("contr.treatment", "contr.poly"))
house.plr <- polr(Sat ~ Infl + Type + Cont, weights = Freq, data = housing)
house.plr
summary(house.plr, digits = 3)
## slightly worse fit from
summary(update(house.plr, method = "probit", Hess = TRUE), digits = 3)
## although it is not really appropriate, can fit
summary(update(house.plr, method = "loglog", Hess = TRUE), digits = 3)
summary(update(house.plr, method = "cloglog", Hess = TRUE), digits = 3)

predict(house.plr, housing, type = "p")
addterm(house.plr, ~.^2, test = "Chisq")
house.plr2 <- stepAIC(house.plr, ~.^2)
house.plr2$anova
anova(house.plr, house.plr2)

house.plr <- update(house.plr, Hess=TRUE)
pr <- profile(house.plr)
confint(pr)
plot(pr)
pairs(pr)
```

---

predict.glmmPQL *Predict Method for glmmPQL Fits*

Description

Obtains predictions from a fitted generalized linear model with random effects.

Usage

```
## S3 method for class 'glmmPQL'
predict(object, newdata = NULL, type = c("link", "response"),
       level, na.action = na.pass, ...)
```

Arguments

object	a fitted object of class inheriting from "glmmPQL".
newdata	optionally, a data frame in which to look for variables with which to predict.
type	the type of prediction required. The default is on the scale of the linear predictors; the alternative "response" is on the scale of the response variable. Thus for a default binomial model the default predictions are of log-odds (probabilities on logit scale) and type = "response" gives the predicted probabilities.
level	an optional integer vector giving the level(s) of grouping to be used in obtaining the predictions. Level values increase from outermost to innermost grouping, with level zero corresponding to the population predictions. Defaults to the highest or innermost level of grouping.
na.action	function determining what should be done with missing values in newdata. The default is to predict NA.
...	further arguments passed to or from other methods.

Value

If level is a single integer, a vector otherwise a data frame.

See Also

glmmPQL, predict.lme.

Examples

```
fit <- glmmPQL(y ~ trt + I(week > 2), random = ~1 |  ID,
               family = binomial, data = bacteria)
predict(fit, bacteria, level = 0, type="response")
predict(fit, bacteria, level = 1, type="response")
```

---

predict.lda *Classify Multivariate Observations by Linear Discrimination*

Description

Classify multivariate observations in conjunction with lda, and also project data onto the linear discriminants.

Usage

```
## S3 method for class 'lda'
predict(object, newdata, prior = object$prior, dimen,
        method = c("plug-in", "predictive", "debiased"), ...)
```

Arguments

object	object of class "lda"
newdata	data frame of cases to be classified or, if object has a formula, a data frame with columns of the same names as the variables used. A vector will be interpreted as a row vector. If newdata is missing, an attempt will be made to retrieve the data used to fit the lda object.
prior	The prior probabilities of the classes, by default the proportions in the training set or what was set in the call to lda.
dimen	the dimension of the space to be used. If this is less than min(p, ng-1), only the first dimen discriminant components are used (except for method="predictive"), and only those dimensions are returned in x.
method	This determines how the parameter estimation is handled. With "plug-in" (the default) the usual unbiased parameter estimates are used and assumed to be correct. With "debiased" an unbiased estimator of the log posterior probabilities is used, and with "predictive" the parameter estimates are integrated out using a vague prior.
...	arguments based from or to other methods

Details

This function is a method for the generic function predict() for class "lda". It can be invoked by calling predict(x) for an object x of the appropriate class, or directly by calling predict.lda(x) regardless of the class of the object.

Missing values in newdata are handled by returning NA if the linear discriminants cannot be evaluated. If newdata is omitted and the na.action of the fit omitted cases, these will be omitted on the prediction.

This version centres the linear discriminants so that the weighted mean (weighted by prior) of the group centroids is at the origin.

Value

a list with components

class	The MAP classification (a factor)
posterior	posterior probabilities for the classes
x	the scores of test cases on up to dimen discriminant variables

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

Ripley, B. D. (1996) *Pattern Recognition and Neural Networks*. Cambridge University Press.

See Also

lda, qda, predict.qda

Examples

```
tr <- sample(1:50, 25)
train <- rbind(iris3[tr,,1], iris3[tr,,2], iris3[tr,,3])
test <- rbind(iris3[-tr,,1], iris3[-tr,,2], iris3[-tr,,3])
cl <- factor(c(rep("s",25), rep("c",25), rep("v",25)))
z <- lda(train, cl)
predict(z, test)$class
```

---

predict.lqs *Predict from an lqs Fit*

Description

Predict from an resistant regression fitted by lqs.

Usage

```
## S3 method for class 'lqs'
predict(object, newdata, na.action = na.pass, ...)
```

Arguments

object	object inheriting from class "lqs"
newdata	matrix or data frame of cases to be predicted or, if object has a formula, a data frame with columns of the same names as the variables used. A vector will be interpreted as a row vector. If newdata is missing, an attempt will be made to retrieve the data used to fit the lqs object.
na.action	function determining what should be done with missing values in newdata. The default is to predict NA.
...	arguments to be passed from or to other methods.

Details

This function is a method for the generic function predict() for class lqs. It can be invoked by calling predict(x) for an object x of the appropriate class, or directly by calling predict.lqs(x) regardless of the class of the object.

Missing values in newdata are handled by returning NA if the linear fit cannot be evaluated. If newdata is omitted and the na.action of the fit omitted cases, these will be omitted on the prediction.

Value

A vector of predictions.

Author(s)

B.D. Ripley

See Also

lqs

Examples

```
set.seed(123)
fm <- lqs(stack.loss ~ ., data = stackloss, method = "S", nsamp = "exact")
predict(fm, stackloss)
```

---

predict.mca *Predict Method for Class 'mca'*

Description

Used to compute coordinates for additional rows or additional factors in a multiple correspondence analysis.

Usage

```
## S3 method for class 'mca'
predict(object, newdata, type = c("row", "factor"), ...)
```

Arguments

object	An object of class "mca", usually the result of a call to mca.
newdata	A data frame containing *either* additional rows of the factors used to fit object *or* additional factors for the cases used in the original fit.
type	Are predictions required for further rows or for new factors?
...	Additional arguments from predict: unused.

Value

If type = "row", the coordinates for the additional rows.

If type = "factor", the coordinates of the column vertices for the levels of the new factors.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

mca, plot.mca

predict.qda *Classify from Quadratic Discriminant Analysis*

Description

Classify multivariate observations in conjunction with qda

Usage

```
## S3 method for class 'qda'
predict(object, newdata, prior = object$prior,
        method = c("plug-in", "predictive", "debiased", "looCV"), ...)
```

Arguments

object	object of class "qda"
newdata	data frame of cases to be classified or, if object has a formula, a data frame with columns of the same names as the variables used. A vector will be interpreted as a row vector. If newdata is missing, an attempt will be made to retrieve the data used to fit the qda object.
prior	The prior probabilities of the classes, by default the proportions in the training set or what was set in the call to qda.
method	This determines how the parameter estimation is handled. With "plug-in" (the default) the usual unbiased parameter estimates are used and assumed to be correct. With "debiased" an unbiased estimator of the log posterior probabilities is used, and with "predictive" the parameter estimates are integrated out using a vague prior. With "looCV" the leave-one-out cross-validation fits to the original dataset are computed and returned.
...	arguments based from or to other methods

Details

This function is a method for the generic function predict() for class "qda". It can be invoked by calling predict(x) for an object x of the appropriate class, or directly by calling predict.qda(x) regardless of the class of the object.

Missing values in newdata are handled by returning NA if the quadratic discriminants cannot be evaluated. If newdata is omitted and the na.action of the fit omitted cases, these will be omitted on the prediction.

Value

a list with components

class	The MAP classification (a factor)
posterior	posterior probabilities for the classes

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

Ripley, B. D. (1996) *Pattern Recognition and Neural Networks*. Cambridge University Press.

See Also

qda, lda, predict.lda

Examples

```
tr <- sample(1:50, 25)
train <- rbind(iris3[tr,,1], iris3[tr,,2], iris3[tr,,3])
test <- rbind(iris3[-tr,,1], iris3[-tr,,2], iris3[-tr,,3])
cl <- factor(c(rep("s",25), rep("c",25), rep("v",25)))
zq <- qda(train, cl)
predict(zq, test)$class
```

---

profile.glm *Method for Profiling glm Objects*

Description

Investigates the profile log-likelihood function for a fitted model of class "glm".

As from R 4.4.0 was migrated to package **stats** with additional functionality.

---

qda *Quadratic Discriminant Analysis*

Description

Quadratic discriminant analysis.

Usage

```
qda(x, ...)

## S3 method for class 'formula'
qda(formula, data, ..., subset, na.action)

## Default S3 method:
qda(x, grouping, prior = proportions,
    method, CV = FALSE, nu, ...)

## S3 method for class 'data.frame'
qda(x, ...)

## S3 method for class 'matrix'
qda(x, grouping, ..., subset, na.action)
```

Arguments

formula	A formula of the form groups ~ x1 + x2 + ... That is, the response is the grouping factor and the right hand side specifies the (non-factor) discriminators.
data	An optional data frame, list or environment from which variables specified in formula are preferentially to be taken.
x	(required if no formula is given as the principal argument.) a matrix or data frame or Matrix containing the explanatory variables.
grouping	(required if no formula principal argument is given.) a factor specifying the class for each observation.
prior	the prior probabilities of class membership. If unspecified, the class proportions for the training set are used. If specified, the probabilities should be specified in the order of the factor levels.
subset	An index vector specifying the cases to be used in the training sample. (NOTE: If given, this argument must be named.)
na.action	A function to specify the action to be taken if NAs are found. The default action is for the procedure to fail. An alternative is na.omit, which leads to rejection of cases with missing values on any required variable. (NOTE: If given, this argument must be named.)
method	"moment" for standard estimators of the mean and variance, "mle" for MLEs, "mve" to use cov.mve, or "t" for robust estimates based on a t distribution.
CV	If true, returns results (classes and posterior probabilities) for leave-out-out cross-validation. Note that if the prior is estimated, the proportions in the whole dataset are used.
nu	degrees of freedom for method = "t".
...	arguments passed to or from other methods.

Details

Uses a QR decomposition which will give an error message if the within-group variance is singular for any group.

Value

an object of class "qda" containing the following components:

prior	the prior probabilities used.
means	the group means.
scaling	for each group i, scaling[,,i] is an array which transforms observations so that within-groups covariance matrix is spherical.
ldet	a vector of half log determinants of the dispersion matrix.
lev	the levels of the grouping factor.
terms	(if formula is a formula) an object of mode expression and class term summarizing the formula.
call	the (matched) function call.

unless CV=TRUE, when the return value is a list with components:

class	The MAP classification (a factor)
posterior	posterior probabilities for the classes

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

Ripley, B. D. (1996) *Pattern Recognition and Neural Networks*. Cambridge University Press.

See Also

predict.qda, lda

Examples

```
tr <- sample(1:50, 25)
train <- rbind(iris3[tr,,1], iris3[tr,,2], iris3[tr,,3])
test <- rbind(iris3[-tr,,1], iris3[-tr,,2], iris3[-tr,,3])
cl <- factor(c(rep("s",25), rep("c",25), rep("v",25)))
z <- qda(train, cl)
predict(z,test)$class
```

---

quine *Absenteeism from School in Rural New South Wales*

Description

The quine data frame has 146 rows and 5 columns. Children from Walgett, New South Wales, Australia, were classified by Culture, Age, Sex and Learner status and the number of days absent from school in a particular school year was recorded.

Usage

```
quine
```

Format

This data frame contains the following columns:

Eth ethnic background: Aboriginal or Not, ("A" or "N").

Sex sex: factor with levels ("F" or "M").

Age age group: Primary ("F0"), or forms "F1," "F2" or "F3".

Lrn learner status: factor with levels Average or Slow learner, ("AL" or "SL").

Days days absent from school in the year.

Source

S. Quine, quoted in Aitkin, M. (1978) The analysis of unbalanced cross classifications (with discussion). *Journal of the Royal Statistical Society series A* **141**, 195–223.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

`Rabbit` *Blood Pressure in Rabbits*

Description

Five rabbits were studied on two occasions, after treatment with saline (control) and after treatment with the $5-HT_3$ antagonist MDL 72222. After each treatment ascending doses of phenylbiguanide were injected intravenously at 10 minute intervals and the responses of mean blood pressure measured. The goal was to test whether the cardiogenic chemoreflex elicited by phenylbiguanide depends on the activation of $5-HT_3$ receptors.

Usage

```
Rabbit
```

Format

This data frame contains 60 rows and the following variables:

`BPchange` change in blood pressure relative to the start of the experiment.

`Dose` dose of Phenylbiguanide in micrograms.

`Run` label of run (`"C1"` to `"C5"`, then `"M1"` to `"M5"`).

`Treatment` placebo or the $5-HT_3$ antagonist MDL 72222.

`Animal` label of animal used (`"R1"` to `"R5"`).

Source

J. Ludbrook (1994) Repeated measurements and multiple comparisons in cardiovascular research. *Cardiovascular Research* **28**, 303–311.
[The numerical data are not in the paper but were supplied by Professor Ludbrook]

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

`rational` *Rational Approximation*

Description

Find rational approximations to the components of a real numeric object using a standard continued fraction method.

Usage

```
rational(x, cycles = 10, max.denominator = 2000, ...)
```

Arguments

`x`	Any object of mode numeric. Missing values are now allowed.
`cycles`	The maximum number of steps to be used in the continued fraction approximation process.
`max.denominator`	An early termination criterion. If any partial denominator exceeds `max.denominator` the continued fraction stops at that point.
`...`	arguments passed to or from other methods.

Details

Each component is first expanded in a continued fraction of the form

`x = floor(x) + 1/(p1 + 1/(p2 + ...)))`

where `p1`, `p2`, ... are positive integers, terminating either at `cycles` terms or when a `pj > max.denominator`. The continued fraction is then re-arranged to retrieve the numerator and denominator as integers and the ratio returned as the value.

Value

A numeric object with the same attributes as `x` but with entries rational approximations to the values. This effectively rounds relative to the size of the object and replaces very small entries by zero.

See Also

`fractions`

Examples

```
X <- matrix(runif(25), 5, 5)
zapsmall(solve(X, X/5)) # print near-zeroes as zero
rational(solve(X, X/5))
```

renumerate *Convert a Formula Transformed by 'denumerate'*

Description

denumerate converts a formula written using the conventions of loglm into one that terms is able to process. renumerate converts it back again to a form like the original.

Usage

```
renumerate(x)
```

Arguments

x	A formula, normally as modified by denumerate.

Details

This is an inverse function to denumerate. It is only needed since terms returns an expanded form of the original formula where the non-marginal terms are exposed. This expanded form is mapped back into a form corresponding to the one that the user originally supplied.

Value

A formula where all variables with names of the form .vn, where n is an integer, converted to numbers, n, as allowed by the formula conventions of loglm.

See Also

denumerate

Examples

```
denumerate(~(1+2+3)^3 + a/b)
## ~ (.v1 + .v2 + .v3)^3 + a/b
renumerate(.Last.value)
## ~ (1 + 2 + 3)^3 + a/b
```

rlm *Robust Fitting of Linear Models*

Description

Fit a linear model by robust regression using an M estimator.

Usage

```
rlm(x, ...)

## S3 method for class 'formula'
rlm(formula, data, weights, ..., subset, na.action,
    method = c("M", "MM", "model.frame"),
    wt.method = c("inv.var", "case"),
    model = TRUE, x.ret = TRUE, y.ret = FALSE, contrasts = NULL)

## Default S3 method:
rlm(x, y, weights, ..., w = rep(1, nrow(x)),
    init = "ls", psi = psi.huber,
    scale.est = c("MAD", "Huber", "proposal 2"), k2 = 1.345,
    method = c("M", "MM"), wt.method = c("inv.var", "case"),
    maxit = 20, acc = 1e-4, test.vec = "resid", lqs.control = NULL)

psi.huber(u, k = 1.345, deriv = 0)
psi.hampel(u, a = 2, b = 4, c = 8, deriv = 0)
psi.bisquare(u, c = 4.685, deriv = 0)
```

Arguments

formula	a formula of the form y ~ x1 + x2 + ...
data	an optional data frame, list or environment from which variables specified in formula are preferentially to be taken.
weights	a vector of prior weights for each case.
subset	An index vector specifying the cases to be used in fitting.
na.action	A function to specify the action to be taken if NAs are found. The 'factory-fresh' default action in R is na.omit, and can be changed by options(na.action=).
x	a matrix or data frame containing the explanatory variables.
y	the response: a vector of length the number of rows of x.
method	currently either M-estimation or MM-estimation or (for the formula method only) find the model frame. MM-estimation is M-estimation with Tukey's bi-weight initialized by a specific S-estimator. See the 'Details' section.
wt.method	are the weights case weights (giving the relative importance of case, so a weight of 2 means there are two of these) or the inverse of the variances, so a weight of two means this error is half as variable?

model	should the model frame be returned in the object?
x.ret	should the model matrix be returned in the object?
y.ret	should the response be returned in the object?
contrasts	optional contrast specifications: see lm.
w	(optional) initial down-weighting for each case.
init	(optional) initial values for the coefficients OR a method to find initial values OR the result of a fit with a coef component. Known methods are "ls" (the default) for an initial least-squares fit using weights w*weights, and "lts" for an unweighted least-trimmed squares fit with 200 samples.
psi	the psi function is specified by this argument. It must give (possibly by name) a function g(x, ..., deriv) that for deriv=0 returns psi(x)/x and for deriv=1 returns psi'(x). Tuning constants will be passed in via ....
scale.est	method of scale estimation: re-scaled MAD of the residuals (default) or Huber's proposal 2 (which can be selected by either "Huber" or "proposal 2").
k2	tuning constant used for Huber proposal 2 scale estimation.
maxit	the limit on the number of IWLS iterations.
acc	the accuracy for the stopping criterion.
test.vec	the stopping criterion is based on changes in this vector.
...	additional arguments to be passed to rlm.default or to the psi function.
lqs.control	An optional list of control values for lqs.
u	numeric vector of evaluation points.
k, a, b, c	tuning constants.
deriv	0 or 1: compute values of the psi function or of its first derivative.

Details

Fitting is done by iterated re-weighted least squares (IWLS).

Psi functions are supplied for the Huber, Hampel and Tukey bisquare proposals as psi.huber, psi.hampel and psi.bisquare. Huber's corresponds to a convex optimization problem and gives a unique solution (up to collinearity). The other two will have multiple local minima, and a good starting point is desirable.

Selecting method = "MM" selects a specific set of options which ensures that the estimator has a high breakdown point. The initial set of coefficients and the final scale are selected by an S-estimator with k0 = 1.548; this gives (for $n \gg p$) breakdown point 0.5. The final estimator is an M-estimator with Tukey's biweight and fixed scale that will inherit this breakdown point provided c > k0; this is true for the default value of c that corresponds to 95% relative efficiency at the normal. Case weights are not supported for method = "MM".

Value

An object of class "rlm" inheriting from "lm". Note that the df.residual component is deliberately set to NA to avoid inappropriate estimation of the residual scale from the residual mean square by "lm" methods.

The additional components not in an lm object are

s	the robust scale estimate used
w	the weights used in the IWLS process
psi	the psi function with parameters substituted
conv	the convergence criteria at each iteration
converged	did the IWLS converge?
wresid	a working residual, weighted for "inv.var" weights only.

Note

Prior to version 7.3-52, offset terms in formula were omitted from fitted and predicted values.

References

P. J. Huber (1981) *Robust Statistics*. Wiley.

F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw and W. A. Stahel (1986) *Robust Statistics: The Approach based on Influence Functions*. Wiley.

A. Marazzi (1993) *Algorithms, Routines and S Functions for Robust Statistics*. Wadsworth & Brooks/Cole.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

lm, lqs.

Examples

```
summary(rlm(stack.loss ~ ., stackloss))
rlm(stack.loss ~ ., stackloss, psi = psi.hampel, init = "lts")
rlm(stack.loss ~ ., stackloss, psi = psi.bisquare)
```

---

rms.curv *Relative Curvature Measures for Non-Linear Regression*

Description

Calculates the root mean square parameter effects and intrinsic relative curvatures, c^{θ} and c^{ι}, for a fitted nonlinear regression, as defined in Bates & Watts, section 7.3, p. 253ff

Usage

```
rms.curv(obj)
```

Arguments

obj	Fitted model object of class "nls". The model must be fitted using the default algorithm.

Details

The method of section 7.3.1 of Bates & Watts is implemented. The function `deriv3` should be used generate a model function with first derivative (gradient) matrix and second derivative (Hessian) array attributes. This function should then be used to fit the nonlinear regression model.

A print method, `print.rms.curv`, prints the `pc` and `ic` components only, suitably annotated.

If either `pc` or `ic` exceeds some threshold (0.3 has been suggested) the curvature is unacceptably high for the planar assumption.

Value

A list of class `rms.curv` with components `pc` and `ic` for parameter effects and intrinsic relative curvatures multiplied by sqrt(F), `ct` and `ci` for c^{θ} and c^{ι} (unmultiplied), and `C` the C-array as used in section 7.3.1 of Bates & Watts.

References

Bates, D. M, and Watts, D. G. (1988) *Nonlinear Regression Analysis and its Applications.* Wiley, New York.

See Also

deriv3

Examples

```
# The treated sample from the Puromycin data
mmcurve <- deriv3(~ Vm * conc/(K + conc), c("Vm", "K"),
                  function(Vm, K, conc) NULL)
Treated <- Puromycin[Puromycin$state == "treated", ]
(Purfit1 <- nls(rate ~ mmcurve(Vm, K, conc), data = Treated,
                start = list(Vm=200, K=0.1)))
rms.curv(Purfit1)
##Parameter effects: c^theta x sqrt(F) = 0.2121
##        Intrinsic: c^iota  x sqrt(F) = 0.092
```

---

rnegbin *Simulate Negative Binomial Variates*

Description

Function to generate random outcomes from a Negative Binomial distribution, with mean `mu` and variance `mu + mu^2/theta`.

Usage

```
rnegbin(n, mu = n, theta = stop("'theta' must be specified"))
```

Arguments

n	If a scalar, the number of sample values required. If a vector, length(n) is the number required and n is used as the mean vector if mu is not specified.
mu	The vector of means. Short vectors are recycled.
theta	Vector of values of the theta parameter. Short vectors are recycled.

Details

The function uses the representation of the Negative Binomial distribution as a continuous mixture of Poisson distributions with Gamma distributed means. Unlike rnbinom the index can be arbitrary.

Value

Vector of random Negative Binomial variate values.

Side Effects

Changes .Random.seed in the usual way.

Examples

```
# Negative Binomials with means fitted(fm) and theta = 4.5
fm <- glm.nb(Days ~ ., data = quine)
dummy <- rnegbin(fitted(fm), theta = 4.5)
```

---

road *Road Accident Deaths in US States*

Description

A data frame with the annual deaths in road accidents for half the US states.

Usage

```
road
```

Format

Columns are:

`state` name.

`deaths` number of deaths.

`drivers` number of drivers (in 10,000s).

`popden` population density in people per square mile.

`rural` length of rural roads, in 1000s of miles.

`temp` average daily maximum temperature in January.

`fuel` fuel consumption in 10,000,000 US gallons per year.

Source

Imperial College, London M.Sc. exercise

`rotifer` *Numbers of Rotifers by Fluid Density*

Description

The data give the numbers of rotifers falling out of suspension for different fluid densities. There are two species, `pm` *Polyartha major* and `kc`, *Keratella cochlearis* and for each species the number falling out and the total number are given.

Usage

```
rotifer
```

Format

`density` specific density of fluid.

`pm.y` number falling out for *P. major*.

`pm.total` total number of *P. major*.

`kc.y` number falling out for *K. cochlearis*.

`kc.tot` total number of *K. cochlearis*.

Source

D. Collett (1991) *Modelling Binary Data.* Chapman & Hall. p. 217

`Rubber` *Accelerated Testing of Tyre Rubber*

Description

Data frame from accelerated testing of tyre rubber.

Usage

```
Rubber
```

Format

`loss` the abrasion loss in gm/hr.

`hard` the hardness in Shore units.

`tens` tensile strength in kg/sq m.

Source

O.L. Davies (1947) *Statistical Methods in Research and Production.* Oliver and Boyd, Table 6.1 p. 119.

O.L. Davies and P.L. Goldsmith (1972) *Statistical Methods in Research and Production.* 4th edition, Longmans, Table 8.1 p. 239.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

`sammon` *Sammon's Non-Linear Mapping*

Description

One form of non-metric multidimensional scaling.

Usage

```
sammon(d, y = cmdscale(d, k), k = 2, niter = 100, trace = TRUE,
       magic = 0.2, tol = 1e-4)
```

Arguments

`d`	distance structure of the form returned by `dist`, or a full, symmetric matrix. Data are assumed to be dissimilarities or relative distances, but must be positive except for self-distance. This can contain missing values.
`y`	An initial configuration. If none is supplied, `cmdscale` is used to provide the classical solution. (If there are missing values in `d`, an initial configuration must be provided.) This must not have duplicates.
`k`	The dimension of the configuration.
`niter`	The maximum number of iterations.
`trace`	Logical for tracing optimization. Default `TRUE`.
`magic`	initial value of the step size constant in diagonal Newton method.
`tol`	Tolerance for stopping, in units of stress.

Details

This chooses a two-dimensional configuration to minimize the stress, the sum of squared differences between the input distances and those of the configuration, weighted by the distances, the whole sum being divided by the sum of input distances to make the stress scale-free.

An iterative algorithm is used, which will usually converge in around 50 iterations. As this is necessarily an $O(n^2)$ calculation, it is slow for large datasets. Further, since the configuration is only determined up to rotations and reflections (by convention the centroid is at the origin), the result can vary considerably from machine to machine. In this release the algorithm has been modified by adding a step-length search (`magic`) to ensure that it always goes downhill.

Value

Two components:

points	A two-column vector of the fitted configuration.
stress	The final stress achieved.

Side Effects

If trace is true, the initial stress and the current stress are printed out every 10 iterations.

References

Sammon, J. W. (1969) A non-linear mapping for data structure analysis. *IEEE Trans. Comput.*, **C-18** 401–409.

Ripley, B. D. (1996) *Pattern Recognition and Neural Networks*. Cambridge University Press.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

cmdscale, isoMDS

Examples

```
swiss.x <- as.matrix(swiss[, -1])
swiss.sam <- sammon(dist(swiss.x))
plot(swiss.sam$points, type = "n")
text(swiss.sam$points, labels = as.character(1:nrow(swiss.x)))
```

---

ships *Ships Damage Data*

Description

Data frame giving the number of damage incidents and aggregate months of service by ship type, year of construction, and period of operation.

Usage

```
ships
```

Format

`type` type: `"A"` to `"E"`.

`year` year of construction: 1960–64, 65–69, 70–74, 75–79 (coded as `"60"`, `"65"`, `"70"`, `"75"`).

`period` period of operation : 1960–74, 75–79.

`service` aggregate months of service.

`incidents` number of damage incidents.

Source

P. McCullagh and J. A. Nelder, (1983), *Generalized Linear Models.* Chapman & Hall, section 6.3.2, page 137

`shoes` *Shoe wear data of Box, Hunter and Hunter*

Description

A list of two vectors, giving the wear of shoes of materials A and B for one foot each of ten boys.

Usage

```
shoes
```

Source

G. E. P. Box, W. G. Hunter and J. S. Hunter (1978) *Statistics for Experimenters.* Wiley, p. 100

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

`shrimp` *Percentage of Shrimp in Shrimp Cocktail*

Description

A numeric vector with 18 determinations by different laboratories of the amount (percentage of the declared total weight) of shrimp in shrimp cocktail.

Usage

```
shrimp
```

Source

F. J. King and J. J. Ryan (1976) Collaborative study of the determination of the amount of shrimp in shrimp cocktail. *J. Off. Anal. Chem.* **59**, 644–649.

R. G. Staudte and S. J. Sheather (1990) *Robust Estimation and Testing.* Wiley.

shuttle *Space Shuttle Autolander Problem*

Description

The shuttle data frame has 256 rows and 7 columns. The first six columns are categorical variables giving example conditions; the seventh is the decision. The first 253 rows are the training set, the last 3 the test conditions.

Usage

```
shuttle
```

Format

This data frame contains the following factor columns:

stability stable positioning or not (stab / xstab).

error size of error (MM / SS / LX / XL).

sign sign of error, positive or negative (pp / nn).

wind wind sign (head / tail).

magn wind strength (Light / Medium / Strong / Out of Range).

vis visibility (yes / no).

use use the autolander or not. (auto / noauto.)

Source

D. Michie (1989) Problems of computer-aided concept formation. In *Applications of Expert Systems 2*, ed. J. R. Quinlan, Turing Institute Press / Addison-Wesley, pp. 310–333.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

Sitka *Growth Curves for Sitka Spruce Trees in 1988*

Description

The Sitka data frame has 395 rows and 4 columns. It gives repeated measurements on the log-size of 79 Sitka spruce trees, 54 of which were grown in ozone-enriched chambers and 25 were controls. The size was measured five times in 1988, at roughly monthly intervals.

Usage

```
Sitka
```

Format

This data frame contains the following columns:

`size` measured size (height times diameter squared) of tree, on log scale.

`Time` time of measurement in days since 1 January 1988.

`tree` number of tree.

`treat` either `"ozone"` for an ozone-enriched chamber or `"control"`.

Source

P. J. Diggle, K.-Y. Liang and S. L. Zeger (1994) *Analysis of Longitudinal Data.* Clarendon Press, Oxford

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

See Also

`Sitka89`.

`Sitka89` *Growth Curves for Sitka Spruce Trees in 1989*

Description

The `Sitka89` data frame has 632 rows and 4 columns. It gives repeated measurements on the log-size of 79 Sitka spruce trees, 54 of which were grown in ozone-enriched chambers and 25 were controls. The size was measured eight times in 1989, at roughly monthly intervals.

Usage

```
Sitka89
```

Format

This data frame contains the following columns:

`size` measured size (height times diameter squared) of tree, on log scale.

`Time` time of measurement in days since 1 January 1988.

`tree` number of tree.

`treat` either `"ozone"` for an ozone-enriched chamber or `"control"`.

Source

P. J. Diggle, K.-Y. Liang and S. L. Zeger (1994) *Analysis of Longitudinal Data.* Clarendon Press, Oxford

See Also

Sitka

Skye *AFM Compositions of Aphyric Skye Lavas*

Description

The Skye data frame has 23 rows and 3 columns.

Usage

```
Skye
```

Format

This data frame contains the following columns:

A Percentage of sodium and potassium oxides.

F Percentage of iron oxide.

M Percentage of magnesium oxide.

Source

R. N. Thompson, J. Esson and A. C. Duncan (1972) Major element chemical variation in the Eocene lavas of the Isle of Skye. *J. Petrology*, **13**, 219–253.

References

J. Aitchison (1986) *The Statistical Analysis of Compositional Data.* Chapman and Hall, p.360.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

Examples

```
# ternary() is from the on-line answers.
ternary <- function(X, pch = par("pch"), lcex = 1,
                    add = FALSE, ord = 1:3, ...)
{
  X <- as.matrix(X)
  if(any(X < 0)) stop("X must be non-negative")
  s <- drop(X %*% rep(1, ncol(X)))
  if(any(s<=0)) stop("each row of X must have a positive sum")
  if(max(abs(s-1)) > 1e-6) {
```

```
    warning("row(s) of X will be rescaled")
    X <- X / s
  }
  X <- X[, ord]
  s3 <- sqrt(1/3)
  if(!add)
  {
    oldpty <- par("pty")
    on.exit(par(pty=oldpty))
    par(pty="s")
    plot(c(-s3, s3), c(0.5-s3, 0.5+s3), type="n", axes=FALSE,
         xlab="", ylab="")
    polygon(c(0, -s3, s3), c(1, 0, 0), density=0)
    lab <- NULL
    if(!is.null(dn <- dimnames(X))) lab <- dn[[2]]
    if(length(lab) < 3) lab <- as.character(1:3)
    eps <- 0.05 * lcex
    text(c(0, s3+eps*0.7, -s3-eps*0.7),
         c(1+eps, -0.1*eps, -0.1*eps), lab, cex=lcex)
  }
  points((X[,2] - X[,3])*s3, X[,1], ...)
}

ternary(Skye/100, ord=c(1,3,2))
```

`snails` *Snail Mortality Data*

Description

Groups of 20 snails were held for periods of 1, 2, 3 or 4 weeks in carefully controlled conditions of temperature and relative humidity. There were two species of snail, A and B, and the experiment was designed as a 4 by 3 by 4 by 2 completely randomized design. At the end of the exposure time the snails were tested to see if they had survived; the process itself is fatal for the animals. The object of the exercise was to model the probability of survival in terms of the stimulus variables, and in particular to test for differences between species.

The data are unusual in that in most cases fatalities during the experiment were fairly small.

Usage

```
snails
```

Format

The data frame contains the following components:

`Species` snail species A (1) or B (2).

`Exposure` exposure in weeks.

`Rel.Hum` relative humidity (4 levels).

`Temp` temperature, in degrees Celsius (3 levels).

`Deaths` number of deaths.

`N` number of snails exposed.

Source

Zoology Department, The University of Adelaide.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

SP500 *Returns of the Standard and Poors 500*

Description

Returns of the Standard and Poors 500 Index in the 1990's

Usage

```
SP500
```

Format

A vector of returns of the Standard and Poors 500 index for all the trading days in 1990, 1991, . . . , 1999.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

---

stdres *Extract Standardized Residuals from a Linear Model*

Description

The standardized residuals. These are normalized to unit variance, fitted including the current data point.

Usage

```
stdres(object)
```

Arguments

`object`	any object representing a linear model.

Value

The vector of appropriately transformed residuals.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

See Also

`residuals`, `studres`

`steam` *The Saturated Steam Pressure Data*

Description

Temperature and pressure in a saturated steam driven experimental device.

Usage

```
steam
```

Format

The data frame contains the following components:

`Temp` temperature, in degrees Celsius.

`Press` pressure, in Pascals.

Source

N.R. Draper and H. Smith (1981) *Applied Regression Analysis.* Wiley, pp. 518–9.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

stepAIC *Choose a model by AIC in a Stepwise Algorithm*

Description

Performs stepwise model selection by AIC.

Usage

```
stepAIC(object, scope, scale = 0,
        direction = c("both", "backward", "forward"),
        trace = 1, keep = NULL, steps = 1000, use.start = FALSE,
        k = 2, ...)
```

Arguments

object	an object representing a model of an appropriate class. This is used as the initial model in the stepwise search.
scope	defines the range of models examined in the stepwise search. This should be either a single formula, or a list containing components upper and lower, both formulae. See the details for how to specify the formulae and how they are used.
scale	used in the definition of the AIC statistic for selecting the models, currently only for lm and aov models (see extractAIC for details).
direction	the mode of stepwise search, can be one of "both", "backward", or "forward", with a default of "both". If the scope argument is missing the default for direction is "backward".
trace	if positive, information is printed during the running of stepAIC. Larger values may give more information on the fitting process.
keep	a filter function whose input is a fitted model object and the associated AIC statistic, and whose output is arbitrary. Typically keep will select a subset of the components of the object and return them. The default is not to keep anything.
steps	the maximum number of steps to be considered. The default is 1000 (essentially as many as required). It is typically used to stop the process early.
use.start	if true the updated fits are done starting at the linear predictor for the currently selected model. This may speed up the iterative calculations for glm (and other fits), but it can also slow them down. **Not used** in R.
k	the multiple of the number of degrees of freedom used for the penalty. Only k = 2 gives the genuine AIC: k = log(n) is sometimes referred to as BIC or SBC.
...	any additional arguments to extractAIC. (None are currently used.)

Details

The set of models searched is determined by the `scope` argument. The right-hand-side of its `lower` component is always included in the model, and right-hand-side of the model is included in the `upper` component. If `scope` is a single formula, it specifies the `upper` component, and the `lower` model is empty. If `scope` is missing, the initial model is used as the `upper` model.

Models specified by `scope` can be templates to update `object` as used by `update.formula`.

There is a potential problem in using `glm` fits with a variable `scale`, as in that case the deviance is not simply related to the maximized log-likelihood. The `glm` method for `extractAIC` makes the appropriate adjustment for a `gaussian` family, but may need to be amended for other cases. (The `binomial` and `poisson` families have fixed `scale` by default and do not correspond to a particular maximum-likelihood problem for variable `scale`.)

Where a conventional deviance exists (e.g. for `lm`, `aov` and `glm` fits) this is quoted in the analysis of variance table: it is the *unscaled* deviance.

Value

the stepwise-selected model is returned, with up to two additional components. There is an `"anova"` component corresponding to the steps taken in the search, as well as a `"keep"` component if the `keep=` argument was supplied in the call. The `"Resid. Dev"` column of the analysis of deviance table refers to a constant minus twice the maximized log likelihood: it will be a deviance only in cases where a saturated model is well-defined (thus excluding `lm`, `aov` and `survreg` fits, for example).

Note

The model fitting must apply the models to the same dataset. This may be a problem if there are missing values and an `na.action` other than `na.fail` is used (as is the default in R). We suggest you remove the missing values first.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

`addterm`, `dropterm`, `step`

Examples

```
quine.hi <- aov(log(Days + 2.5) ~ .^4, quine)
quine.nxt <- update(quine.hi, . ~ . - Eth:Sex:Age:Lrn)
quine.stp <- stepAIC(quine.nxt,
    scope = list(upper = ~Eth*Sex*Age*Lrn, lower = ~1),
    trace = FALSE)
quine.stp$anova

cpus1 <- cpus
for(v in names(cpus)[2:7])
  cpus1[[v]] <- cut(cpus[[v]], unique(quantile(cpus[[v]])),
```

```
                    include.lowest = TRUE)
cpus0 <- cpus1[, 2:8]  # excludes names, authors' predictions
cpus.samp <- sample(1:209, 100)
cpus.lm <- lm(log10(perf) ~ ., data = cpus1[cpus.samp,2:8])
cpus.lm2 <- stepAIC(cpus.lm, trace = FALSE)
cpus.lm2$anova

example(birthwt)
birthwt.glm <- glm(low ~ ., family = binomial, data = bwt)
birthwt.step <- stepAIC(birthwt.glm, trace = FALSE)
birthwt.step$anova
birthwt.step2 <- stepAIC(birthwt.glm, ~ .^2 + I(scale(age)^2)
    + I(scale(lwt)^2), trace = FALSE)
birthwt.step2$anova

quine.nb <- glm.nb(Days ~ .^4, data = quine)
quine.nb2 <- stepAIC(quine.nb)
quine.nb2$anova
```

`stormer` *The Stormer Viscometer Data*

Description

The stormer viscometer measures the viscosity of a fluid by measuring the time taken for an inner cylinder in the mechanism to perform a fixed number of revolutions in response to an actuating weight. The viscometer is calibrated by measuring the time taken with varying weights while the mechanism is suspended in fluids of accurately known viscosity. The data comes from such a calibration, and theoretical considerations suggest a nonlinear relationship between time, weight and viscosity, of the form `Time = (B1*Viscosity)/(Weight - B2) + E` where `B1` and `B2` are unknown parameters to be estimated, and `E` is error.

Usage

```
stormer
```

Format

The data frame contains the following components:

`Viscosity` viscosity of fluid.

`Wt` actuating weight.

`Time` time taken.

Source

E. J. Williams (1959) *Regression Analysis.* Wiley.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

`studres` *Extract Studentized Residuals from a Linear Model*

Description

The Studentized residuals. Like standardized residuals, these are normalized to unit variance, but the Studentized version is fitted ignoring the current data point. (They are sometimes called jackknifed residuals).

Usage

```
studres(object)
```

Arguments

`object`	any object representing a linear model.

Value

The vector of appropriately transformed residuals.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

`residuals`, `stdres`

`summary.loglm` *Summary Method Function for Objects of Class 'loglm'*

Description

Returns a summary list for log-linear models fitted by iterative proportional scaling using `loglm`.

Usage

```
## S3 method for class 'loglm'
summary(object, fitted = FALSE, ...)
```

Arguments

object	a fitted loglm model object.
fitted	if TRUE return observed and expected frequencies in the result. Using fitted = TRUE may necessitate re-fitting the object.
...	arguments to be passed to or from other methods.

Details

This function is a method for the generic function summary() for class "loglm". It can be invoked by calling summary(x) for an object x of the appropriate class, or directly by calling summary.loglm(x) regardless of the class of the object.

Value

a list is returned for use by print.summary.loglm. This has components

formula	the formula used to produce object
tests	the table of test statistics (likelihood ratio, Pearson) for the fit.
oe	if fitted = TRUE, an array of the observed and expected frequencies, otherwise NULL.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

loglm, summary

---

summary.negbin *Summary Method Function for Objects of Class 'negbin'*

Description

Identical to summary.glm, but with three lines of additional output: the ML estimate of theta, its standard error, and twice the log-likelihood function.

Usage

```
## S3 method for class 'negbin'
summary(object, dispersion = 1, correlation = FALSE, ...)
```

Arguments

object	fitted model object of class negbin inheriting from glm and lm. Typically the output of glm.nb.
dispersion	as for summary.glm, with a default of 1.
correlation	as for summary.glm.
...	arguments passed to or from other methods.

Details

summary.glm is used to produce the majority of the output and supply the result. This function is a method for the generic function summary() for class "negbin". It can be invoked by calling summary(x) for an object x of the appropriate class, or directly by calling summary.negbin(x) regardless of the class of the object.

Value

As for summary.glm; the additional lines of output are not included in the resultant object.

Side Effects

A summary table is produced as for summary.glm, with the additional information described above.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

summary, glm.nb, negative.binomial, anova.negbin

Examples

```
## IGNORE_RDIFF_BEGIN
summary(glm.nb(Days ~ Eth*Age*Lrn*Sex, quine, link = log))
## IGNORE_RDIFF_END
```

---

summary.rlm *Summary Method for Robust Linear Models*

Description

summary method for objects of class "rlm"

Usage

```
## S3 method for class 'rlm'
summary(object, method = c("XtX", "XtWX"), correlation = FALSE, ...)
```

Arguments

object	the fitted model. This is assumed to be the result of some fit that produces an object inheriting from the class rlm, in the sense that the components returned by the rlm function will be available.
method	Should the weighted (by the IWLS weights) or unweighted cross-products matrix be used?
correlation	logical. Should correlations be computed (and printed)?
...	arguments passed to or from other methods.

Details

This function is a method for the generic function summary() for class "rlm". It can be invoked by calling summary(x) for an object x of the appropriate class, or directly by calling summary.rlm(x) regardless of the class of the object.

Value

If printing takes place, only a null value is returned. Otherwise, a list is returned with the following components. Printing always takes place if this function is invoked automatically as a method for the summary function.

correlation	The computed correlation coefficient matrix for the coefficients in the model.
cov.unscaled	The unscaled covariance matrix; i.e, a matrix such that multiplying it by an estimate of the error variance produces an estimated covariance matrix for the coefficients.
sigma	The scale estimate.
stddev	A scale estimate used for the standard errors.
df	The number of degrees of freedom for the model and for residuals.
coefficients	A matrix with three columns, containing the coefficients, their standard errors and the corresponding t statistic.
terms	The terms object used in fitting this model.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

summary

Examples

```
summary(rlm(calls ~ year, data = phones, maxit = 50))
```

survey *Student Survey Data*

Description

This data frame contains the responses of 237 Statistics I students at the University of Adelaide to a number of questions.

Usage

```
survey
```

Format

The components of the data frame are:

`Sex` The sex of the student. (Factor with levels `"Male"` and `"Female"`.)

`Wr.Hnd` span (distance from tip of thumb to tip of little finger of spread hand) of writing hand, in centimetres.

`NW.Hnd` span of non-writing hand.

`W.Hnd` writing hand of student. (Factor, with levels `"Left"` and `"Right"`.)

`Fold` "Fold your arms! Which is on top" (Factor, with levels `"R on L"`, `"L on R"`, `"Neither"`.)

`Pulse` pulse rate of student (beats per minute).

`Clap` 'Clap your hands! Which hand is on top?' (Factor, with levels `"Right"`, `"Left"`, `"Neither"`.)

`Exer` how often the student exercises. (Factor, with levels `"Freq"` (frequently), `"Some"`, `"None"`.)

`Smoke` how much the student smokes. (Factor, levels `"Heavy"`, `"Regul"` (regularly), `"Occas"` (occasionally), `"Never"`.)

`Height` height of the student in centimetres.

`M.I` whether the student expressed height in imperial (feet/inches) or metric (centimetres/metres) units. (Factor, levels `"Metric"`, `"Imperial"`.)

`Age` age of the student in years.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

synth.tr *Synthetic Classification Problem*

Description

The `synth.tr` data frame has 250 rows and 3 columns. The `synth.te` data frame has 100 rows and 3 columns. It is intended that `synth.tr` be used from training and `synth.te` for testing.

Usage

```
synth.tr
synth.te
```

Format

These data frames contains the following columns:

`xs` x-coordinate

`ys` y-coordinate

`yc` class, coded as 0 or 1.

Source

Ripley, B.D. (1994) Neural networks and related methods for classification (with discussion). *Journal of the Royal Statistical Society series B* **56**, 409–456.

Ripley, B.D. (1996) *Pattern Recognition and Neural Networks.* Cambridge: Cambridge University Press.

theta.md *Estimate theta of the Negative Binomial*

Description

Given the estimated mean vector, estimate `theta` of the Negative Binomial Distribution.

Usage

```
theta.md(y, mu, dfr, weights, limit = 20, eps = .Machine$double.eps^0.25)

theta.ml(y, mu, n, weights, limit = 10, eps = .Machine$double.eps^0.25,
         trace = FALSE)

theta.mm(y, mu, dfr, weights, limit = 10, eps = .Machine$double.eps^0.25)
```

Arguments

y	Vector of observed values from the Negative Binomial.
mu	Estimated mean vector.
n	Number of data points (defaults to the sum of weights)
dfr	Residual degrees of freedom (assuming theta known). For a weighted fit this is the sum of the weights minus the number of fitted parameters.
weights	Case weights. If missing, taken as 1.
limit	Limit on the number of iterations.
eps	Tolerance to determine convergence.
trace	logical: should iteration progress be printed?

Details

theta.md estimates by equating the deviance to the residual degrees of freedom, an analogue of a moment estimator.

theta.ml uses maximum likelihood.

theta.mm calculates the moment estimator of theta by equating the Pearson chi-square $\sum(y - \mu)^2/(\mu + \mu^2/\theta)$ to the residual degrees of freedom.

Value

The required estimate of theta, as a scalar. For theta.ml, the standard error is given as attribute "SE".

See Also

glm.nb

Examples

```
quine.nb <- glm.nb(Days ~ .^2, data = quine)
theta.md(quine$Days, fitted(quine.nb), dfr = df.residual(quine.nb))
theta.ml(quine$Days, fitted(quine.nb))
theta.mm(quine$Days, fitted(quine.nb), dfr = df.residual(quine.nb))

## weighted example
yeast <- data.frame(cbind(numbers = 0:5, fr = c(213, 128, 37, 18, 3, 1)))
fit <- glm.nb(numbers ~ 1, weights = fr, data = yeast)
## IGNORE_RDIFF_BEGIN
summary(fit)
## IGNORE_RDIFF_END
mu <- fitted(fit)
theta.md(yeast$numbers, mu, dfr = 399, weights = yeast$fr)
theta.ml(yeast$numbers, mu, limit = 15, weights = yeast$fr)
theta.mm(yeast$numbers, mu, dfr = 399, weights = yeast$fr)
```

topo *Spatial Topographic Data*

Description

The topo data frame has 52 rows and 3 columns, of topographic heights within a 310 feet square.

Usage

```
topo
```

Format

This data frame contains the following columns:

x x coordinates (units of 50 feet)

y y coordinates (units of 50 feet)

z heights (feet)

Source

Davis, J.C. (1973) *Statistics and Data Analysis in Geology.* Wiley.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

Traffic *Effect of Swedish Speed Limits on Accidents*

Description

An experiment was performed in Sweden in 1961–2 to assess the effect of a speed limit on the motorway accident rate. The experiment was conducted on 92 days in each year, matched so that day j in 1962 was comparable to day j in 1961. On some days the speed limit was in effect and enforced, while on other days there was no speed limit and cars tended to be driven faster. The speed limit days tended to be in contiguous blocks.

Usage

```
Traffic
```

Format

This data frame contains the following columns:

`year` 1961 or 1962.

`day` of year.

`limit` was there a speed limit?

`y` traffic accident count for that day.

Source

Svensson, A. (1981) On the goodness-of-fit test for the multiplicative Poisson model. *Annals of Statistics,* **9**, 697–704.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

---

truehist *Plot a Histogram*

Description

Creates a histogram on the current graphics device.

Usage

```
truehist(data, nbins = "Scott", h, x0 = -h/1000,
         breaks, prob = TRUE, xlim = range(breaks),
         ymax = max(est), col = "cyan",
         xlab = deparse(substitute(data)), bty = "n", ...)
```

Arguments

`data`	numeric vector of data for histogram. Missing values (`NA`s) are allowed and omitted.
`nbins`	The suggested number of bins. Either a positive integer, or a character string naming a rule: `"Scott"` or `"Freedman-Diaconis"` or `"FD"`. (Case is ignored.)
`h`	The bin width, a strictly positive number (takes precedence over `nbins`).
`x0`	Shift for the bins - the breaks are at `x0 + h * (..., -1, 0, 1, ...)`
`breaks`	The set of breakpoints to be used. (Usually omitted, takes precedence over `h` and `nbins`).

prob	If true (the default) plot a true histogram. The vertical axis has a *relative frequency density* scale, so the product of the dimensions of any panel gives the relative frequency. Hence the total area under the histogram is 1 and it is directly comparable with most other estimates of the probability density function. If false plot the counts in the bins.
xlim	The limits for the x-axis.
ymax	The upper limit for the y-axis.
col	The colour for the bar fill: the default is colour 5 in the default R palette.
xlab	label for the plot x-axis. By default, this will be the name of data.
bty	The box type for the plot - defaults to none.
...	additional arguments to rect or plot.

Details

This plots a true histogram, a density estimate of total area 1. If breaks is specified, those breakpoints are used. Otherwise if h is specified, a regular grid of bins is used with width h. If neither breaks nor h is specified, nbins is used to select a suitable h.

Side Effects

A histogram is plotted on the current device.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

hist

---

ucv — *Unbiased Cross-Validation for Bandwidth Selection*

Description

Uses unbiased cross-validation to select the bandwidth of a Gaussian kernel density estimator.

Usage

```
ucv(x, nb = 1000, lower, upper)
```

Arguments

x	a numeric vector
nb	number of bins to use.
lower, upper	Range over which to minimize. The default is almost always satisfactory.

Value

a bandwidth.

References

Scott, D. W. (1992) *Multivariate Density Estimation: Theory, Practice, and Visualization.* Wiley.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

See Also

`bcv`, `width.SJ`, `density`

Examples

```
ucv(geyser$duration)
```

UScereal *Nutritional and Marketing Information on US Cereals*

Description

The `UScereal` data frame has 65 rows and 11 columns. The data come from the 1993 ASA Statistical Graphics Exposition, and are taken from the mandatory F&DA food label. The data have been normalized here to a portion of one American cup.

Usage

```
UScereal
```

Format

This data frame contains the following columns:

`mfr` Manufacturer, represented by its first initial: G=General Mills, K=Kelloggs, N=Nabisco, P=Post, Q=Quaker Oats, R=Ralston Purina.

`calories` number of calories in one portion.

`protein` grams of protein in one portion.

`fat` grams of fat in one portion.

`sodium` milligrams of sodium in one portion.

`fibre` grams of dietary fibre in one portion.

`carbo` grams of complex carbohydrates in one portion.

`sugars` grams of sugars in one portion.

`shelf` display shelf (1, 2, or 3, counting from the floor).

`potassium` grams of potassium.

`vitamins` vitamins and minerals (none, enriched, or 100%).

Source

The original data are available at http://lib.stat.cmu.edu/datasets/1993.expo/.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

UScrime *The Effect of Punishment Regimes on Crime Rates*

Description

Criminologists are interested in the effect of punishment regimes on crime rates. This has been studied using aggregate data on 47 states of the USA for 1960 given in this data frame. The variables seem to have been re-scaled to convenient numbers.

Usage

```
UScrime
```

Format

This data frame contains the following columns:

`M` percentage of males aged 14–24.

`So` indicator variable for a Southern state.

`Ed` mean years of schooling.

`Po1` police expenditure in 1960.

`Po2` police expenditure in 1959.

`LF` labour force participation rate.

`M.F` number of males per 1000 females.

`Pop` state population.

`NW` number of non-whites per 1000 people.

`U1` unemployment rate of urban males 14–24.

`U2` unemployment rate of urban males 35–39.

`GDP` gross domestic product per head.

`Ineq` income inequality.

`Prob` probability of imprisonment.

`Time` average time served in state prisons.

`y` rate of crimes in a particular category per head of population.

Source

Ehrlich, I. (1973) Participation in illegitimate activities: a theoretical and empirical investigation. *Journal of Political Economy*, **81**, 521–565.

Vandaele, W. (1978) Participation in illegitimate activities: Ehrlich revisited. In *Deterrence and Incapacitation*, eds A. Blumstein, J. Cohen and D. Nagin, pp. 270–335. US National Academy of Sciences.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S-PLUS.* Fourth Edition. Springer.

VA *Veteran's Administration Lung Cancer Trial*

Description

Veteran's Administration lung cancer trial from Kalbfleisch & Prentice.

Usage

```
VA
```

Format

A data frame with columns:

`stime` survival or follow-up time in days.

`status` dead or censored.

`treat` treatment: standard or test.

`age` patient's age in years.

`Karn` Karnofsky score of patient's performance on a scale of 0 to 100.

`diag.time` times since diagnosis in months at entry to trial.

`cell` one of four cell types.

`prior` prior therapy?

Source

Kalbfleisch, J.D. and Prentice R.L. (1980) *The Statistical Analysis of Failure Time Data.* Wiley.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

`waders` *Counts of Waders at 15 Sites in South Africa*

Description

The `waders` data frame has 15 rows and 19 columns. The entries are counts of waders in summer.

Usage

```
waders
```

Format

This data frame contains the following columns (species)

`S1` Oystercatcher
`S2` White-fronted Plover
`S3` Kitt Lutz's Plover
`S4` Three-banded Plover
`S5` Grey Plover
`S6` Ringed Plover
`S7` Bar-tailed Godwit
`S8` Whimbrel
`S9` Marsh Sandpiper
`S10` Greenshank
`S11` Common Sandpiper
`S12` Turnstone
`S13` Knot
`S14` Sanderling
`S15` Little Stint
`S16` Curlew Sandpiper
`S17` Ruff
`S18` Avocet
`S19` Black-winged Stilt

The rows are the sites:

A = Namibia North coast
B = Namibia North wetland
C = Namibia South coast
D = Namibia South wetland
E = Cape North coast
F = Cape North wetland

G = Cape West coast
H = Cape West wetland
I = Cape South coast
J= Cape South wetland
K = Cape East coast
L = Cape East wetland
M = Transkei coast
N = Natal coast
O = Natal wetland

Source

J.C. Gower and D.J. Hand (1996) *Biplots* Chapman & Hall Table 9.1. Quoted as from:

R.W. Summers, L.G. Underhill, D.J. Pearson and D.A. Scott (1987) Wader migration systems in south and eastern Africa and western Asia. *Wader Study Group Bulletin* **49** Supplement, 15–34.

Examples

```
plot(corresp(waders, nf=2))
```

---

whiteside *House Insulation: Whiteside's Data*

Description

Mr Derek Whiteside of the UK Building Research Station recorded the weekly gas consumption and average external temperature at his own house in south-east England for two heating seasons, one of 26 weeks before, and one of 30 weeks after cavity-wall insulation was installed. The object of the exercise was to assess the effect of the insulation on gas consumption.

Usage

```
whiteside
```

Format

The `whiteside` data frame has 56 rows and 3 columns.:

`Insul` A factor, before or after insulation.

`Temp` Purportedly the average outside temperature in degrees Celsius. (These values is far too low for any 56-week period in the 1960s in South-East England. It might be the weekly average of daily minima.)

`Gas` The weekly gas consumption in 1000s of cubic feet.

Source

A data set collected in the 1960s by Mr Derek Whiteside of the UK Building Research Station. Reported by

Hand, D. J., Daly, F., McConway, K., Lunn, D. and Ostrowski, E. eds (1993) *A Handbook of Small Data Sets.* Chapman & Hall, p. 69.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

Examples

```
require(lattice)
xyplot(Gas ~ Temp | Insul, whiteside, panel =
  function(x, y, ...) {
    panel.xyplot(x, y, ...)
    panel.lmline(x, y, ...)
  }, xlab = "Average external temperature (deg. C)",
  ylab = "Gas consumption  (1000 cubic feet)", aspect = "xy",
  strip = function(...) strip.default(..., style = 1))

gasB <- lm(Gas ~ Temp, whiteside, subset = Insul=="Before")
gasA <- update(gasB, subset = Insul=="After")
summary(gasB)
summary(gasA)
gasBA <- lm(Gas ~ Insul/Temp - 1, whiteside)
summary(gasBA)

gasQ <- lm(Gas ~ Insul/(Temp + I(Temp^2)) - 1, whiteside)
coef(summary(gasQ))

gasPR <- lm(Gas ~ Insul + Temp, whiteside)
anova(gasPR, gasBA)
options(contrasts = c("contr.treatment", "contr.poly"))
gasBA1 <- lm(Gas ~ Insul*Temp, whiteside)
coef(summary(gasBA1))
```

---

width.SJ *Bandwidth Selection by Pilot Estimation of Derivatives*

Description

Uses the method of Sheather & Jones (1991) to select the bandwidth of a Gaussian kernel density estimator.

Usage

```
width.SJ(x, nb = 1000, lower, upper, method = c("ste", "dpi"))
```

Arguments

x	a numeric vector
nb	number of bins to use.
upper, lower	range over which to search for solution if method = "ste".
method	Either "ste" ("solve-the-equation") or "dpi" ("direct plug-in").

Value

a bandwidth.

Note

A faster version for large n (thousands) is available in R $\geq$ 3.4.0 as part of bw.SJ: quadruple its value for comparability with this version.

References

Sheather, S. J. and Jones, M. C. (1991) A reliable data-based bandwidth selection method for kernel density estimation. *Journal of the Royal Statistical Society series B* **53**, 683–690.

Scott, D. W. (1992) *Multivariate Density Estimation: Theory, Practice, and Visualization.* Wiley.

Wand, M. P. and Jones, M. C. (1995) *Kernel Smoothing.* Chapman & Hall.

See Also

ucv, bcv, density

Examples

```
width.SJ(geyser$duration, method = "dpi")
width.SJ(geyser$duration)

width.SJ(galaxies, method = "dpi")
width.SJ(galaxies)
```

---

write.matrix *Write a Matrix or Data Frame*

Description

Writes a matrix or data frame to a file or the console, using column labels and a layout respecting columns.

Usage

```
write.matrix(x, file = "", sep = " ", blocksize)
```

Arguments

x	matrix or data frame.
file	name of output file. The default ("") is the console.
sep	The separator between columns.
blocksize	If supplied and positive, the output is written in blocks of blocksize rows. Choose as large as possible consistent with the amount of memory available.

Details

If x is a matrix, supplying blocksize is more memory-efficient and enables larger matrices to be written, but each block of rows might be formatted slightly differently.

If x is a data frame, the conversion to a matrix may negate the memory saving.

Side Effects

A formatted file is produced, with column headings (if x has them) and columns of data.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

See Also

write.table

---

wtloss *Weight Loss Data from an Obese Patient*

Description

The data frame gives the weight, in kilograms, of an obese patient at 52 time points over an 8 month period of a weight rehabilitation programme.

Usage

```
wtloss
```

Format

This data frame contains the following columns:

Days time in days since the start of the programme.

Weight weight in kilograms of the patient.

Source

Dr T. Davies, Adelaide.

References

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. Springer.

Examples

```
## IGNORE_RDIFF_BEGIN
wtloss.fm <- nls(Weight ~ b0 + b1*2^(-Days/th),
    data = wtloss, start = list(b0=90, b1=95, th=120))
wtloss.fm
## IGNORE_RDIFF_END
plot(wtloss)
with(wtloss, lines(Days, fitted(wtloss.fm)))
```

Index