

Package ‘L1pack’

November 14, 2017

Version 0.38.19

Date 2017-11-14

Title Routines for L1 Estimation

Description L1 estimation for linear regression, density, distribution function, quantile function and random number generation for univariate and multivariate Laplace distribution.

Author Felipe Osorio [aut, cre], Tymoteusz Wolodzko [aut]

Maintainer Felipe Osorio <felipe.osorios@usm.cl>

Imports stats, grDevices, graphics

Suggests heavy

License GPL (>= 2.0)

URL <http://l1pack.mat.utfsm.cl>

LazyLoad yes

NeedsCompilation yes

Repository CRAN

Date/Publication 2017-11-14 20:54:16 UTC

R topics documented:

l1fit	2
l1pack.control	3
lad	3
Laplace	5
rmLaplace	6
simulate.lad	7
Index	9

`l1fit`*Minimum absolute residual (L1) regression*

Description

Performs an L1 regression on a matrix of explanatory variables and a vector of responses.

Usage

```
l1fit(x, y, intercept = TRUE, tolerance = 1e-07, print.it = TRUE)
```

Arguments

<code>x</code>	vector or matrix of explanatory variables. Each row corresponds to an observation and each column to a variable. The number of rows of <code>x</code> should equal the number of data values in <code>y</code> , and there should be fewer columns than rows. Missing values are not allowed.
<code>y</code>	numeric vector containing the response. Missing values are not allowed.
<code>intercept</code>	logical flag. If TRUE, an intercept term is included in the regression model.
<code>tolerance</code>	numerical value used to test for singularity in the regression.
<code>print.it</code>	logical flag. If TRUE, then warnings about non-unique solutions and rank deficiency are given.

Details

The Barrodale-Roberts algorithm, which is a specialized linear programming algorithm, is used.

Value

list defining the regression (compare with function `lsfit`).

<code>coefficients</code>	vector of coefficients.
<code>residuals</code>	residuals from the fit.
<code>message</code>	vector of one or two character strings stating whether a non-unique solution is possible, or if the <code>x</code> matrix was found to be rank deficient.

References

Barrodale, I., and Roberts, F.D.K. (1973). An improved algorithm for discrete L1 linear approximations. *SIAM Journal of Numerical Analysis* **10**, 839-848.

Barrodale, I., and Roberts, F.D.K. (1974). Solution of an overdetermined system of equations in the L1 norm. *Communications of the ACM* **17**, 319-320.

Bloomfield, P., and Steiger, W.L. (1983). *Least Absolute Deviations: Theory, Applications, and Algorithms*. Birkhauser, Boston, Mass.

Examples

```
l1fit(stack.x, stack.loss)
```

l1pack.control	<i>Set control parameters</i>
----------------	-------------------------------

Description

Allows users to set parameters for [lad](#).

Usage

```
l1pack.control(maxIter = 2000, tolerance = 1e-9)
```

Arguments

maxIter	maximum number of iterations. The default is 2000.
tolerance	the relative tolerance in the iterative algorithm.

Value

A list of control arguments to be used in a call to [lad](#).

A call to `l1pack.control` can be used directly in the control argument of a call to [lad](#).

Author(s)

Felipe Osorio.

Examples

```
ctrl <- l1pack.control(maxIter = 50, tol = 1e-07)
lad(stack.loss ~ ., data = stackloss, control = ctrl)
```

lad	<i>Least absolute deviations regression</i>
-----	---

Description

This function is used to fit linear models considering Laplace errors.

Usage

```
lad(formula, data, method = c("BR", "EM"), subset, na.action,
    control, model = TRUE, x = FALSE, y = FALSE, contrasts = NULL)
```

Arguments

formula	an object of class "formula": a symbolic description of the model to be fitted.
data	an optional data frame containing the variables in the model. If not found in data, the variables are taken from <code>environment(formula)</code> , typically the environment from which <code>lad</code> is called.
method	character string specifying the algorithm to use. The default algorithm is the Barrodale and Roberts algorithm <code>method = "BR"</code> . Other possible value is <code>method = "EM"</code> for an EM algorithm using IRLS.
subset	an optional expression indicating the subset of the rows of data that should be used in the fit.
na.action	a function that indicates what should happen when the data contain NAs.
control	a list of control values for the estimation algorithm to replace the default values returned by the function <code>l1pack.control</code> .
model, x, y	logicals. If TRUE the corresponding components of the fit (the model frame, the model matrix, the response) are returned.
contrasts	an optional list. See the <code>contrasts.arg</code> of <code>model.matrix.default</code> .

Value

an object of class `lad` representing the linear model fit. Generic function `print`, show the results of the fit.

The functions `print` and `summary` are used to obtain and print a summary of the results. The generic accessor functions `coefficients`, `fitted.values` and `residuals` extract various useful features of the value returned by `lad`.

Author(s)

The design was inspired by the R function `lm`.

References

- Barrodale, I., and Roberts, F.D.K. (1974). Solution of an overdetermined system of equations in the L1 norm. *Communications of the ACM* **17**, 319-320.
- Phillips, R.F. (2002). Least absolute deviations estimation via the EM algorithm. *Statistics and Computing* **12**, 281-285.

Examples

```
fm <- lad(stack.loss ~ ., data = stackloss, method = "BR")
summary(fm)
```

Laplace

*The Laplace distribution***Description**

Density, distribution function, quantile function and random generation for the Laplace distribution with location parameter `location` and scale parameter `scale`.

Usage

```
dlaplace(x, location = 0, scale = 1, log = FALSE)
plaplace(q, location = 0, scale = 1, lower.tail = TRUE, log.p = FALSE)
qlaplace(p, location = 0, scale = 1, lower.tail = TRUE, log.p = FALSE)
rlaplace(n, location = 0, scale = 1)
```

Arguments

`x, q` vector of quantiles.
`location, scale` location and scale parameters. Scale must be positive.
`log, log.p` logical; if TRUE, probabilities `p` are given as $\log(p)$.
`lower.tail` logical; if TRUE (default), probabilities are $P[X \leq x]$, otherwise, $P[X > x]$.
`p` vector of probabilities.
`n` number of observations. If $\text{length}(n) > 1$, the length is taken to be the number required.

Details

If `location` or `scale` are not specified, they assume the default values of 0 and 1 respectively.

The Laplace distribution with location μ and scale ϕ has density

$$f(x) = \frac{1}{\sqrt{2}\phi} \exp(-\sqrt{2}|x - \mu|/\phi)$$

Value

`dlaplace`, `plaplace`, and `qlaplace` are respectively the density, distribution function and quantile function of the Laplace distribution. `rlaplace` generates random deviates from the Laplace.

The length of the result is determined by `n` for `rlaplace`, and is the maximum of the lengths of the numerical parameters for the other functions.

Author(s)

Felipe Osorio and Tymoteusz Wolodzko

References

Kotz, S., Kozubowski, T.J., and Podgorski, K. (2001). *The Laplace Distributions and Generalizations*. Birkhauser, Boston.

Phillips, R.F. (2002). Least absolute deviations estimation via the EM algorithm. *Statistics and Computing* **12**, 281-285.

See Also

[Distributions](#) for other standard distributions and [rmLaplace](#) for the random generation from the multivariate Laplace distribution.

Examples

```
x <- rlaplace(1000)
## Q-Q plot for Laplace data against true theoretical distribution:
qqplot(qlaplace(ppoints(1000)), x, main = "Laplace Q-Q plot",
       xlab = "Theoretical quantiles", ylab = "Sample quantiles")
abline(c(0,1), col = "red", lwd = 2)
```

 rmLaplace

Multivariate Laplace Random Deviates

Description

Random number generation from the multivariate Laplace distribution.

Usage

```
rmLaplace(n = 1, center = rep(0, nrow(Scatter)), Scatter = diag(length(center)))
```

Arguments

n	the number of samples requested
center	a vector giving the locations of each variable
Scatter	a positive-definite dispersion matrix

Details

The function `rmLaplace` is an interface to C routines, which make calls to subroutines from LAPACK. The matrix decomposition is internally done using the Cholesky decomposition. If `Scatter` is not non-negative definite then there will be a warning message.

Value

If `n = 1` a vector of the same length as `center`, otherwise a matrix of `n` rows of random vectors.

References

Gomez, E., Gomez-Villegas, M.A., and Marin, J.M. (1998). A multivariate generalization of the power exponential family of distributions. *Communications in Statistics - Theory and Methods* **27**, 589-600.

Kotz, S., Kozubowski, T.J., and Podgorski, K. (2001). *The Laplace Distributions and Generalizations*. Birkhauser, Boston.

Examples

```
# dispersion parameters
Scatter <- matrix(c(1,.5,.5,1), ncol = 2)
Scatter

# generate the sample
y <- rmLaplace(n = 2000, Scatter = Scatter)

# scatterplot of a random bivariate Laplace sample with center
# vector zero and scale matrix 'Scatter'
par(pty = "s")
plot(y, xlab = "", ylab = "")
title("bivariate Laplace sample", font.main = 1)
```

simulate.lad

Simulate Responses from lad Models

Description

Simulate one or more responses from the distribution corresponding to a fitted lad object.

Usage

```
## S3 method for class 'lad'
simulate(object, nsim = 1, seed = NULL, ...)
```

Arguments

object	an object representing a fitted model.
nsim	number of response vectors to simulate. Defaults to 1.
seed	an object specifying if and how the random number generator should be initialized ('seeded'). For the "lad" method, either NULL or an integer that will be used in a call to <code>set.seed</code> before simulating the response vectors. If set, the value is saved as the "seed" attribute of the returned value. The default, NULL will not change the random generator state, and return <code>.Random.seed</code> as the "seed" attribute, see 'Value'.
...	additional optional arguments.

Value

For the "lad" method, the result is a data frame with an attribute "seed". If argument seed is NULL, the attribute is the value of [.Random.seed](#) before the simulation was started.

Author(s)

Tymoteusz Wolodzko and Felipe Osorio

Examples

```
fm <- lad(stack.loss ~ ., data = stackloss)
sm <- simulate(fm, nsim = 4)
```


Index

- *Topic **datagen**
 - simulate.lad, 7
- *Topic **distribution**
 - Laplace, 5
 - rmlaplace, 6
- *Topic **models**
 - l1pack.control, 3
 - simulate.lad, 7
- *Topic **multivariate**
 - rmlaplace, 6
- *Topic **regression**
 - l1fit, 2
 - l1pack.control, 3
 - lad, 3
- .Random.seed, 7, 8
- Distributions, 6
- dlaplace (Laplace), 5
- l1fit, 2
- l1pack.control, 3, 4
- lad, 3, 3
- Laplace, 5
- lm, 4
- lsfit, 2
- plaplace (Laplace), 5
- qlaplace (Laplace), 5
- rlaplace (Laplace), 5
- rmlaplace, 6, 6
- simulate.lad, 7