

# Package ‘logspline’

July 22, 2025

**Version** 2.1.22

**Date** 2024-05-10

**Maintainer** Charles Kooperberg <clk@fredhutch.org>

**Title** Routines for Logspline Density Estimation

**Description** Contains routines for logspline density estimation.

The function `oldlogspline()` uses the same algorithm as the logspline package version 1.0.x; i.e. the Kooperberg and Stone (1992) algorithm (with an improved interface). The recommended routine `logspline()` uses an algorithm from Stone et al (1997) <[DOI:10.1214/aos/1031594728](https://doi.org/10.1214/aos/1031594728)>.

**Imports** stats, graphics

**License** Apache License 2.0

**NeedsCompilation** yes

**Author** Charles Kooperberg [aut, cre],  
Cleve Moler [ctb] (LINPACK routines in src),  
Jack Dongarra [ctb] (LINPACK routines in src)

**Repository** CRAN

**Date/Publication** 2024-05-10 21:13:10 UTC

## Contents

<code>dlogspline</code> . . . . .	2
<code>doldlogspline</code> . . . . .	3
<code>logspline</code> . . . . .	4
<code>oldlogspline</code> . . . . .	7
<code>oldlogspline.to.logspline</code> . . . . .	9
<code>plot.logspline</code> . . . . .	10
<code>plot.oldlogspline</code> . . . . .	12
<code>summary.logspline</code> . . . . .	13
<code>summary.oldlogspline</code> . . . . .	14
<code>unstrip</code> . . . . .	15

<b>Index</b>	17
--------------	----

dlogspline

*Log spline Density Estimation***Description**

Density (dlogspline), cumulative probability (plogspline), quantiles (qlogspline), and random samples (rlogspline) from a logspline density that was fitted using the 1997 knot addition and deletion algorithm ([logspline](#)). The 1992 algorithm is available using the [oldlogspline](#) function.

**Usage**

```
dlogspline(q, fit, log = FALSE)
plogspline(q, fit)
qlogspline(p, fit)
rlogspline(n, fit)
```

**Arguments**

q	vector of quantiles. Missing values (NAs) are allowed.
p	vector of probabilities. Missing values (NAs) are allowed.
n	sample size. If length(n) is larger than 1, then length(n) random values are returned.
fit	logspline object, typically the result of <a href="#">logspline</a> .
log	should dlogspline return densities (TRUE) or log-densities (FALSE)

**Details**

Elements of q or p that are missing will cause the corresponding elements of the result to be missing.

**Value**

Densities (dlogspline), probabilities (plogspline), quantiles (qlogspline), or a random sample (rlogspline) from a logspline density that was fitted using knot addition and deletion.

**Author(s)**

Charles Kooperberg <clk@fredhutch.org>.

**References**

Charles Kooperberg and Charles J. Stone. Log spline density estimation for censored data (1992). *Journal of Computational and Graphical Statistics*, **1**, 301–328.

Charles J. Stone, Mark Hansen, Charles Kooperberg, and Young K. Truong. The use of polynomial splines and their tensor products in extended linear modeling (with discussion) (1997). *Annals of Statistics*, **25**, 1371–1470.

**See Also**

[logspline](#), [plot.logspline](#), [summary.logspline](#), [oldlogspline](#).

**Examples**

```
x <- rnorm(100)
fit <- logspline(x)
qq <- qlogspline((1:99)/100, fit)
plot(qnorm((1:99)/100), qq)           # qq plot of the fitted density
pp <- plogspline((-250:250)/100, fit)
plot((-250:250)/100, pp, type = "l")
lines((-250:250)/100, pnorm((-250:250)/100)) # asses the fit of the distribution
dd <- dlogspline((-250:250)/100, fit)
plot((-250:250)/100, dd, type = "l")
lines((-250:250)/100, dnorm((-250:250)/100)) # asses the fit of the density
rr <- rlogspline(100, fit)           # random sample from fit
```

---

doldlogspline

*Logspline Density Estimation - 1992 version*


---

**Description**

Probability density function (`doldlogspline`), distribution function (`poldlogspline`), quantiles (`qoldlogspline`), and random samples (`roldlogspline`) from a logspline density that was fitted using the 1992 knot deletion algorithm ([oldlogspline](#)). The 1997 algorithm using knot deletion and addition is available using the [logspline](#) function.

**Usage**

```
doldlogspline(q, fit)
poldlogspline(q, fit)
qoldlogspline(p, fit)
roldlogspline(n, fit)
```

**Arguments**

<code>q</code>	vector of quantiles. Missing values (NAs) are allowed.
<code>p</code>	vector of probabilities. Missing values (NAs) are allowed.
<code>n</code>	sample size. If <code>length(n)</code> is larger than 1, then <code>length(n)</code> random values are returned.
<code>fit</code>	<code>oldlogspline</code> object, typically the result of <a href="#">oldlogspline</a> .

**Details**

Elements of `q` or `p` that are missing will cause the corresponding elements of the result to be missing.

**Value**

Densities (`doldlogspline`), probabilities (`poldlogspline`), quantiles (`qoldlogspline`), or a random sample (`roldlogspline`) from an `oldlogspline` density that was fitted using knot deletion.

**Author(s)**

Charles Kooperberg <clk@fredhutch.org>.

**References**

Charles Kooperberg and Charles J. Stone. Logspline density estimation for censored data (1992). *Journal of Computational and Graphical Statistics*, **1**, 301–328.

Charles J. Stone, Mark Hansen, Charles Kooperberg, and Young K. Truong. The use of polynomial splines and their tensor products in extended linear modeling (with discussion) (1997). *Annals of Statistics*, **25**, 1371–1470.

**See Also**

[logspline](#), [oldlogspline](#), [plot.oldlogspline](#), [summary.oldlogspline](#)

**Examples**

```
x <- rnorm(100)
fit <- oldlogspline(x)
qq <- qoldlogspline((1:99)/100, fit)
plot(qnorm((1:99)/100), qq)           # qq plot of the fitted density
pp <- poldlogspline((-250:250)/100, fit)
plot((-250:250)/100, pp, type = "l")
lines((-250:250)/100, pnorm((-250:250)/100)) # asses the fit of the distribution
dd <- doldlogspline((-250:250)/100, fit)
plot((-250:250)/100, dd, type = "l")
lines((-250:250)/100, dnorm((-250:250)/100)) # asses the fit of the density
rr <- roldlogspline(100, fit)          # random sample from fit
```

---

logspline

*Logspline Density Estimation*


---

**Description**

Fits a logspline density using splines to approximate the log-density using the 1997 knot addition and deletion algorithm ([logspline](#)). The 1992 algorithm is available using the [oldlogspline](#) function.

**Usage**

```
logspline(x, lbound, ubound, maxknots = 0, knots, nknots = 0, penalty,
  silent = TRUE, mind = -1, error.action = 2)
```

**Arguments**

<code>x</code>	data vector. The data needs to be uncensored. <a href="#">oldlogspline</a> can deal with right- left- and interval-censored data.
<code>lbound, ubound</code>	lower/upper bound for the support of the density. For example, if there is a priori knowledge that the density equals zero to the left of 0, and has a discontinuity at 0, the user could specify <code>lbound = 0</code> . However, if the density is essentially zero near 0, one does not need to specify <code>lbound</code> .
<code>maxknots</code>	the maximum number of knots. The routine stops adding knots when this number of knots is reached. The method has an automatic rule for selecting max-knots if this parameter is not specified.
<code>knots</code>	ordered vector of values (that should cover the complete range of the observations), which forces the method to start with these knots. Overrides <code>knots</code> . If <code>knots</code> is not specified, a default knot-placement rule is employed.
<code>nknots</code>	forces the method to start with <code>nknots</code> knots. The method has an automatic rule for selecting <code>nknots</code> if this parameter is not specified.
<code>penalty</code>	the parameter to be used in the AIC criterion. The method chooses the number of knots that minimizes $-2 * \text{loglikelihood} + \text{penalty} * (\text{number of knots} - 1)$ . The default is to use a penalty parameter of <code>penalty = log(samplesize)</code> as in BIC. The effect of this parameter is summarized in <a href="#">summary.logspline</a> .
<code>silent</code>	should diagnostic output be printed?
<code>mind</code>	minimum distance, in order statistics, between knots.
<code>error.action</code>	how should logspline deal with non-convergence problems? Very-very rarely in some extreme situations logspline has convergence problems. The only two situations that I am aware of are when there is effectively a sharp bound, but this bound was not specified, or when the data is severely rounded. logspline can deal with this in three ways. If <code>error.action</code> is 2, the same data is re-run with the slightly more stable, but less flexible <code>oldlogspline</code> . The object is translated in a logspline object using <code>oldlogspline.to.logspline</code> , so this is almost invisible to the user. It is particularly useful when you run simulation studies, as the code can seamlessly continue. Only the <code>lbound</code> and <code>ubound</code> options are passed on to <code>oldlogspline</code> , other options revert to the default. If <code>error.action</code> is 1, a warning is printed, and logspline returns nothing (but does not crash). This is useful if you run a simulation, but do not like to revert to <code>oldlogspline</code> . If <code>error.action</code> is 0, the code crashes using the stop function.

**Value**

Object of the class `logspline`, that is intended as input for [plot.logspline](#) (summary plots), [summary.logspline](#) (fitting summary), [dlogspline](#) (densities), [plogspline](#) (probabilities), [qlogspline](#) (quantiles), [rlogspline](#) (random numbers from the fitted distribution).

The object has the following members:

<code>call</code>	the command that was executed.
<code>nknots</code>	the number of knots in the model that was selected.

<code>coef.pol</code>	coefficients of the polynomial part of the spline. The first coefficient is the constant term and the second is the linear term.
<code>coef.kts</code>	coefficients of the knots part of the spline. The $k$ -th element is the coefficient of $(x - t(k))_+^3$ (where $x_+^3$ means the positive part of the third power of $x$ , and $t(k)$ means knot $k$ ).
<code>knots</code>	vector of the locations of the knots in the logspline model.
<code>maxknots</code>	the largest number of knots minus one considered during fitting (i.e. with <code>maxknots</code> = 6 the maximum number of knots is 5).
<code>penalty</code>	the penalty that was used.
<code>bound</code>	first element: 0 - lbound was $-\infty$ , 1 it was something else; second element: lbound, if specified; third element: 0 - ubound was $\infty$ , 1 it was something else; fourth element: ubound, if specified.
<code>samples</code>	the sample size.
<code>logl</code>	matrix with 3 columns. Column one: number of knots; column two: model fitted during addition (1) or deletion (2); column 3: log-likelihood.
<code>range</code>	range of the input data.
<code>mind</code>	minimum distance in order statistics between knots required during fitting (the actual minimum distance may be much larger).

### Author(s)

Charles Kooperberg <clk@fredhutch.org>.

### References

Charles Kooperberg and Charles J. Stone. Logspline density estimation for censored data (1992). *Journal of Computational and Graphical Statistics*, **1**, 301–328.

Charles J. Stone, Mark Hansen, Charles Kooperberg, and Young K. Truong. The use of polynomial splines and their tensor products in extended linear modeling (with discussion) (1997). *Annals of Statistics*, **25**, 1371–1470.

### See Also

[plot.logspline](#), [summary.logspline](#), [dlogspline](#), [plogspline](#), [qlogspline](#), [rlogspline](#), [oldlogspline](#), [oldlogspline.to.logspline](#).

### Examples

```
y <- rnorm(100)
fit <- logspline(y)
plot(fit)
#
# as (4 == length(-2, -1, 0, 1, 2) - 1), this forces these initial knots,
# and does no knot selection
fit <- logspline(y, knots = c(-2, -1, 0, 1, 2), maxknots = 4, penalty = 0)
#
# the following example give one of the rare examples where logspline
```

```
# crashes, and this shows the use of error.action = 2.
#
set.seed(118)
zz <- rnorm(300)
zz[151:300] <- zz[151:300]+5
zz <- round(zz)
fit <- logspline(zz)
#
# you could rerun this with
# fit <- logspline(zz, error.action=0)
# or
# fit <- logspline(zz, error.action=1)
```

---

oldlogspline

*Logspline Density Estimation - 1992 version*


---

## Description

Fits a logspline density using splines to approximate the log-density using the 1992 knot deletion algorithm ([oldlogspline](#)). The 1997 algorithm using knot deletion and addition is available using the [logspline](#) function.

## Usage

```
oldlogspline(uncensored, right, left, interval, lbound,
ubound, nknots, knots, penalty, delete = TRUE)
```

## Arguments

uncensored	vector of uncensored observations from the distribution whose density is to be estimated. If there are no uncensored observations, this argument can be omitted. However, either uncensored or interval must be specified.
right	vector of right censored observations from the distribution whose density is to be estimated. If there are no right censored observations, this argument can be omitted.
left	vector of left censored observations from the distribution whose density is to be estimated. If there are no left censored observations, this argument can be omitted.
interval	two column matrix of lower and upper bounds of observations that are interval censored from the distribution whose density is to be estimated. If there are no interval censored observations, this argument can be omitted.
lbound, ubound	lower/upper bound for the support of the density. For example, if there is a priori knowledge that the density equals zero to the left of 0, and has a discontinuity at 0, the user could specify <code>lbound = 0</code> . However, if the density is essentially zero near 0, one does not need to specify <code>lbound</code> . The default for <code>lbound</code> is <code>-inf</code> and the default for <code>ubound</code> is <code>inf</code> .

nknots	forces the method to start with nknots knots (delete = TRUE) or to fit a density with nknots knots (delete = FALSE). The method has an automatic rule for selecting nknots if this parameter is not specified.
knots	ordered vector of values (that should cover the complete range of the observations), which forces the method to start with these knots (delete = TRUE) or to fit a density with these knots delete = FALSE). Overrides nknots. If knots is not specified, a default knot-placement rule is employed.
penalty	the parameter to be used in the AIC criterion. The method chooses the number of knots that minimizes $-2 * \text{loglikelihood} + \text{penalty} * (\text{number of knots} - 1)$ . The default is to use a penalty parameter of $\text{penalty} = \log(\text{samplesize})$ as in BIC. The effect of this parameter is summarized in <a href="#">summary.oldlogspline</a> .
delete	should stepwise knot deletion be employed?

### Value

Object of the class `oldlogspline`, that is intended as input for [plot.oldlogspline](#), [summary.oldlogspline](#), [doldlogspline](#) (densities), [poldlogspline](#) (probabilities), [qoldlogspline](#) (quantiles), [roldlogspline](#) (random numbers from the fitted distribution). The function [oldlogspline.to.logspline](#) can translate an object of the class `oldlogspline` to an object of the class `logspline`.

The object has the following members:

call	the command that was executed.
knots	vector of the locations of the knots in the <code>oldlogspline</code> model. <code>old</code>
coef	coefficients of the spline. The first coefficient is the constant term, the second is the linear term and the $k$ -th ( $k > 2$ ) is the coefficient of $(x - t(k - 2))_+^3$ (where $x_+^3$ means the positive part of the third power of $x$ , and $t(k - 2)$ means knot $k - 2$ ). If a coefficient is zero the corresponding knot was deleted from the model.
bound	first element: 0 - lbound was $-\inf$ , 1 it was something else; second element: lbound, if specified; third element: 0 - ubound was $\inf$ , 1 it was something else; fourth element: ubound, if specified.
logl	the $k$ -th element is the log-likelihood of the fit with $k+2$ knots.
penalty	the penalty that was used.
sample	the sample size that was used.
delete	was stepwise knot deletion employed?

### Author(s)

Charles Kooperberg <clk@fredhutch.org>.

### References

Charles Kooperberg and Charles J. Stone. Logsplines density estimation for censored data (1992). *Journal of Computational and Graphical Statistics*, **1**, 301–328.



Charles J. Stone, Mark Hansen, Charles Kooperberg, and Young K. Truong. The use of polynomial splines and their tensor products in extended linear modeling (with discussion) (1997). *Annals of Statistics*, **25**, 1371–1470.

### See Also

[logspline](#), [oldlogspline](#), [plot.oldlogspline](#), [summary.oldlogspline](#), [doldlogspline](#), [poldlogspline](#), [qoldlogspline](#), [roldlogspline](#), [oldlogspline.to.logspline](#).

### Examples

```
# A simple example
y <- rnorm(100)
fit <- oldlogspline(y)
plot(fit)
# An example involving censoring and a lower bound
y <- rlnorm(1000)
censoring <- rexp(1000) * 4
delta <- 1 * (y <= censoring)
y[delta == 0] <- censoring[delta == 0]
fit <- oldlogspline(y[delta == 1], y[delta == 0], lbound = 0)
```

---

oldlogspline.to.logspline

*Logspline Density Estimation - 1992 to 1997 version*

---

### Description

Translates an oldlogspline object in an logspline object. This routine is mostly used in logspline, as it allows the routine to use oldlogspline for some situations where logspline crashes. The other use is when you have censored data, and thus have to use oldlogspline to fit, but wish to use the auxiliary routines from logspline.

### Usage

```
oldlogspline.to.logspline(obj, data)
```

### Arguments

obj	object of class logspline
data	the original data. Used to compute the range component of the new object. if data is not available, the $1/(n+1)$ and $n/(n+1)$ quantiles of the fitted distribution are used for range.

### Value

object of the class logspline. The call component of the new object is not useful. The delete component of the old object is ignored.

**Author(s)**

Charles Kooperberg <clk@fredhutch.org>.

**References**

Charles Kooperberg and Charles J. Stone. Logspline density estimation for censored data (1992). *Journal of Computational and Graphical Statistics*, **1**, 301–328.

Charles J. Stone, Mark Hansen, Charles Kooperberg, and Young K. Truong. The use of polynomial splines and their tensor products in extended linear modeling (with discussion) (1997). *Annals of Statistics*, **25**, 1371–1470.

**See Also**

[logspline](#), [oldlogspline](#).

**Examples**

```
x <- rnorm(100)
fit.old <- oldlogspline(x)
fit.translate <- oldlogspline.to.logspline(fit.old,x)
fit.new <- logspline(x)
plot(fit.new)
plot(fit.old,add=TRUE,col=2)
#
# should look almost the same, the differences are the
# different fitting routines
#
```

---

plot.logspline

*Logspline Density Estimation*

---

**Description**

Plots a logspline density, distribution function, hazard function or survival function from a logspline density that was fitted using the 1997 knot addition and deletion algorithm ([logspline](#)). The 1992 algorithm is available using the [oldlogspline](#) function.

**Usage**

```
## S3 method for class 'logspline'
plot(x, n = 100, what = "d", add = FALSE, xlim, xlab = "",
      ylab = "", type = "l", ...)
```

**Arguments**

x	logspline object, typically the result of <a href="#">logspline</a> .
n	the number of equally spaced points at which to plot the density.
what	what should be plotted: "d" (density), "p" (distribution function), "s" (survival function) or "h" (hazard function).
add	should the plot be added to an existing plot.
xlim	range of data on which to plot. Default is from the 1th to the 99th percentile of the density, extended by 10% on each end.
xlab, ylab	labels plotted on the axes.
type	type of plot.
...	other plotting options, as desired

**Details**

This function produces a plot of a [logspline](#) fit at n equally spaced points roughly covering the support of the density. (Use `xlim = c(from, to)` to change the range of these points.)

**Author(s)**

Charles Kooperberg <clk@fredhutch.org>.

**References**

Charles Kooperberg and Charles J. Stone. Logsplines density estimation for censored data (1992). *Journal of Computational and Graphical Statistics*, **1**, 301–328.

Charles J. Stone, Mark Hansen, Charles Kooperberg, and Young K. Truong. The use of polynomial splines and their tensor products in extended linear modeling (with discussion) (1997). *Annals of Statistics*, **25**, 1371–1470.

**See Also**

[logspline](#), [summary.logspline](#), [dlogspline](#), [plogspline](#), [qlogspline](#), [rlogspline](#), [oldlogspline](#).

**Examples**

```
y <- rnorm(100)
fit <- logspline(y)
plot(fit)
```

---

plot.oldlogspline      *Logspline Density Estimation - 1992 version*


---

## Description

Plots an oldlogspline density, distribution function, hazard function or survival function from a logspline density that was fitted using the 1992 knot deletion algorithm. The 1997 algorithm using knot deletion and addition is available using the [logspline](#) function.

## Usage

```
## S3 method for class 'oldlogspline'
plot(x, n = 100, what = "d", xlim, xlab = "", ylab = "",
     type = "l", add = FALSE, ...)
```

## Arguments

x	logspline object, typically the result of <a href="#">logspline</a> .
n	the number of equally spaced points at which to plot the density.
what	what should be plotted: "d" (density), "p" (distribution function), "s" (survival function) or "h" (hazard function).
xlim	range of data on which to plot. Default is from the 1th to the 99th percentile of the density, extended by 10% on each end.
xlab, ylab	labels plotted on the axes.
type	type of plot.
add	should the plot be added to an existing plot.
...	other plotting options, as desired

## Details

This function produces a plot of a [oldlogspline](#) fit at n equally spaced points roughly covering the support of the density. (Use `xlim=c(from, to)` to change the range of these points.)

## Author(s)

Charles Kooperberg <clk@fredhutch.org>.

## References

Charles Kooperberg and Charles J. Stone. Logspline density estimation for censored data (1992). *Journal of Computational and Graphical Statistics*, **1**, 301–328.

Charles J. Stone, Mark Hansen, Charles Kooperberg, and Young K. Truong. The use of polynomial splines and their tensor products in extended linear modeling (with discussion) (1997). *Annals of Statistics*, **25**, 1371–1470.

**See Also**

[logspline](#), [oldlogspline](#), [summary.oldlogspline](#), [doldlogspline](#), [poldlogspline](#), [qoldlogspline](#), [roldlogspline](#).

**Examples**

```
y <- rnorm(100)
fit <- oldlogspline(y)
plot(fit)
```

---

summary.logspline	<i>Logspline Density Estimation</i>
-------------------	-------------------------------------

---

**Description**

This function summarizes both the stepwise selection process of the model fitting by [logspline](#), as well as the final model that was selected using AIC/BIC. A logspline object was fit using the 1997 knot addition and deletion algorithm. The 1992 algorithm is available using the [oldlogspline](#) function.

**Usage**

```
## S3 method for class 'logspline'
summary(object, ...)
## S3 method for class 'logspline'
print(x, ...)
```

**Arguments**

object, x	logspline object, typically the result of <a href="#">logspline</a>
...	other arguments are ignored.

**Details**

These function produce identical printed output. The main body is a table with five columns: the first column is a possible number of knots for the fitted model; the second column is the log-likelihood for the fit; the third column is  $-2 * \text{loglikelihood} + \text{penalty} * (\text{number of knots} - 1)$ , which is the AIC criterion; [logspline](#) selected the model with the smallest value of AIC; the fourth and fifth columns give the endpoints of the interval of values of penalty that would yield the model with the indicated number of knots. (NAs imply that the model is not optimal for any choice of penalty.) At the bottom of the table the number of knots corresponding to the selected model is reported, as is the value of penalty that was used.

**Author(s)**

Charles Kooperberg <clk@fredhutch.org>.

## References

Charles Kooperberg and Charles J. Stone. Logsplines density estimation for censored data (1992). *Journal of Computational and Graphical Statistics*, **1**, 301–328.

Charles J. Stone, Mark Hansen, Charles Kooperberg, and Young K. Truong. The use of polynomial splines and their tensor products in extended linear modeling (with discussion) (1997). *Annals of Statistics*, **25**, 1371–1470.

## See Also

[logspline](#), [plot.logspline](#), [dlogspline](#), [plogspline](#), [qlogspline](#), [rlogspline](#), [oldlogspline](#).

## Examples

```
y <- rnorm(100)
fit <- logspline(y)
summary(fit)
```

---

summary.oldlogspline    *Logsplines Density Estimation - 1992 version*

---

## Description

This function summarizes both the stepwise selection process of the model fitting by [oldlogspline](#), as well as the final model that was selected using AIC/BIC. A logspline object was fit using the 1992 knot deletion algorithm ([oldlogspline](#)). The 1997 algorithm using knot deletion and addition is available using the [logspline](#) function.

## Usage

```
## S3 method for class 'oldlogspline'
summary(object, ...)
## S3 method for class 'oldlogspline'
print(x, ...)
```

## Arguments

object, x	oldlogspline object, typically the result of <a href="#">oldlogspline</a>
...	other arguments are ignored.

## Details

These function produces the same printed output. The main body is a table with five columns: the first column is a possible number of knots for the fitted model;  
the second column is the log-likelihood for the fit;

the third column is  $-2 * \text{loglikelihood} + \text{penalty} * (\text{number of knots} - 1)$ , which is the AIC criterion; [logspline](#) selected the model with the smallest value of AIC;

the fourth and fifth columns give the endpoints of the interval of values of penalty that would yield the model with the indicated number of knots. (NAs imply that the model is not optimal for any choice of penalty.) At the bottom of the table the number of knots corresponding to the selected model is reported, as is the value of penalty that was used.

### Author(s)

Charles Kooperberg <clk@fredhutch.org>.

### References

Charles Kooperberg and Charles J. Stone. Logspline density estimation for censored data (1992). *Journal of Computational and Graphical Statistics*, **1**, 301–328.

Charles J. Stone, Mark Hansen, Charles Kooperberg, and Young K. Truong. The use of polynomial splines and their tensor products in extended linear modeling (with discussion) (1997). *Annals of Statistics*, **25**, 1371–1470.

### See Also

[logspline](#), [oldlogspline](#), [plot.oldlogspline](#), [doldlogspline](#), [poldlogspline](#), [qoldlogspline](#), [roldlogspline](#).

### Examples

```
y <- rnorm(100)
fit <- oldlogspline(y)
summary(fit)
```

---

unstrip

*Reformat data as vector or matrix*


---

### Description

This function tries to convert a data.frame or a matrix to a no-frills matrix without labels, and a vector or time-series to a no-frills vector without labels.

### Usage

```
unstrip(x)
```

### Arguments

x                      one- or two-dimensional object.

**Details**

Many of the functions for [logspline](#), [oldlogspline](#), were written in the “before data.frame” era; `unstrip` attempts to keep all these functions useful with more advanced input objects. In particular, many of these functions call `unstrip` before doing anything else.

**Value**

If `x` is two-dimensional a matrix without names, if `x` is one-dimensional a numerical vector

**Author(s)**

Charles Kooperberg <clk@fredhutch.org>.

**Examples**

```
data(co2)
unstrip(co2)
data(iris)
unstrip(iris)
```



# Index

- \* **classes**
  - unstrip, [15](#)
- \* **distribution**
  - dlogspline, [2](#)
  - doldlogspline, [3](#)
  - logspline, [4](#)
  - oldlogspline, [7](#)
  - oldlogspline.to.logspline, [9](#)
  - plot.logspline, [10](#)
  - plot.oldlogspline, [12](#)
  - summary.logspline, [13](#)
  - summary.oldlogspline, [14](#)
- \* **smooth**
  - dlogspline, [2](#)
  - doldlogspline, [3](#)
  - logspline, [4](#)
  - oldlogspline, [7](#)
  - oldlogspline.to.logspline, [9](#)
  - plot.logspline, [10](#)
  - plot.oldlogspline, [12](#)
  - summary.logspline, [13](#)
  - summary.oldlogspline, [14](#)
- \* **utilities**
  - unstrip, [15](#)

dlogspline, [2](#), [5](#), [6](#), [11](#), [14](#)  
doldlogspline, [3](#), [8](#), [9](#), [13](#), [15](#)  
  
logspline, [2–4](#), [4](#), [7](#), [9–16](#)  
  
oldlogspline, [2–7](#), [7](#), [9–16](#)  
oldlogspline.to.logspline, [6](#), [8](#), [9](#), [9](#)  
  
plogspline, [5](#), [6](#), [11](#), [14](#)  
plogspline (dlogspline), [2](#)  
plot.logspline, [3](#), [5](#), [6](#), [10](#), [14](#)  
plot.oldlogspline, [4](#), [8](#), [9](#), [12](#), [15](#)  
poldlogspline, [8](#), [9](#), [13](#), [15](#)  
poldlogspline (doldlogspline), [3](#)  
print.logspline (summary.logspline), [13](#)  
  
print.oldlogspline  
    (summary.oldlogspline), [14](#)  
  
qlogspline, [5](#), [6](#), [11](#), [14](#)  
qlogspline (dlogspline), [2](#)  
qoldlogspline, [8](#), [9](#), [13](#), [15](#)  
qoldlogspline (doldlogspline), [3](#)  
  
rlogspline, [5](#), [6](#), [11](#), [14](#)  
rlogspline (dlogspline), [2](#)  
roldlogspline, [8](#), [9](#), [13](#), [15](#)  
roldlogspline (doldlogspline), [3](#)  
  
summary.logspline, [3](#), [5](#), [6](#), [11](#), [13](#)  
summary.oldlogspline, [4](#), [8](#), [9](#), [13](#), [14](#)  
  
unstrip, [15](#)