|  |  |
| --- | --- |
| lm {stats} | R Documentation |

**Fitting Linear Models**

**Description**

lm is used to fit linear models. It can be used to carry out regression, single stratum analysis of variance and analysis of covariance (although [aov](http://127.0.0.1:14695/library/stats/help/aov) may provide a more convenient interface for these).

**Usage**

lm(formula, data, subset, weights, na.action,

method = "qr", model = TRUE, x = FALSE, y = FALSE, qr = TRUE,

singular.ok = TRUE, contrasts = NULL, offset, ...)

**Arguments**

|  |  |
| --- | --- |
| formula | an object of class "[formula](http://127.0.0.1:14695/library/stats/help/formula)" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are given under ‘Details’. |
| data | an optional data frame, list or environment (or object coercible by [as.data.frame](http://127.0.0.1:14695/library/stats/help/as.data.frame) to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula), typically the environment from which lm is called. |
| subset | an optional vector specifying a subset of observations to be used in the fitting process. |
| weights | an optional vector of weights to be used in the fitting process. Should be NULL or a numeric vector. If non-NULL, weighted least squares is used with weights weights (that is, minimizing sum(w\*e^2)); otherwise ordinary least squares is used. See also ‘Details’, |
| na.action | a function which indicates what should happen when the data contain NAs. The default is set by the na.action setting of [options](http://127.0.0.1:14695/library/stats/help/options), and is [na.fail](http://127.0.0.1:14695/library/stats/help/na.fail) if that is unset. The ‘factory-fresh’ default is [na.omit](http://127.0.0.1:14695/library/stats/help/na.omit). Another possible value is NULL, no action. Value [na.exclude](http://127.0.0.1:14695/library/stats/help/na.exclude) can be useful. |
| method | the method to be used; for fitting, currently only method = "qr" is supported; method = "model.frame" returns the model frame (the same as with model = TRUE, see below). |
| model, x, y, qr | logicals. If TRUE the corresponding components of the fit (the model frame, the model matrix, the response, the QR decomposition) are returned. |
| singular.ok | logical. If FALSE (the default in S but not in **R**) a singular fit is an error. |
| contrasts | an optional list. See the contrasts.arg of [model.matrix.default](http://127.0.0.1:14695/library/stats/help/model.matrix.default). |
| offset | this can be used to specify an *a priori* known component to be included in the linear predictor during fitting. This should be NULL or a numeric vector of length equal to the number of cases. One or more [offset](http://127.0.0.1:14695/library/stats/help/offset) terms can be included in the formula instead or as well, and if more than one are specified their sum is used. See [model.offset](http://127.0.0.1:14695/library/stats/help/model.offset). |
| ... | additional arguments to be passed to the low level regression fitting functions (see below). |

**Details**

Models for lm are specified symbolically. A typical model has the form response ~ terms where response is the (numeric) response vector and terms is a series of terms which specifies a linear predictor for response. A terms specification of the form first + second indicates all the terms in first together with all the terms in second with duplicates removed. A specification of the form first:second indicates the set of terms obtained by taking the interactions of all terms in first with all terms in second. The specification first\*second indicates the *cross* of first and second. This is the same as first + second + first:second.

If the formula includes an [offset](http://127.0.0.1:14695/library/stats/help/offset), this is evaluated and subtracted from the response.

If response is a matrix a linear model is fitted separately by least-squares to each column of the matrix.

See [model.matrix](http://127.0.0.1:14695/library/stats/help/model.matrix) for some further details. The terms in the formula will be re-ordered so that main effects come first, followed by the interactions, all second-order, all third-order and so on: to avoid this pass a terms object as the formula (see [aov](http://127.0.0.1:14695/library/stats/help/aov) and demo(glm.vr) for an example).

A formula has an implied intercept term. To remove this use either y ~ x - 1 or y ~ 0 + x. See [formula](http://127.0.0.1:14695/library/stats/help/formula) for more details of allowed formulae.

Non-NULL weights can be used to indicate that different observations have different variances (with the values in weights being inversely proportional to the variances); or equivalently, when the elements of weights are positive integers *w\_i*, that each response *y\_i* is the mean of *w\_i* unit-weight observations (including the case that there are *w\_i* observations equal to *y\_i* and the data have been summarized).

lm calls the lower level functions [lm.fit](http://127.0.0.1:14695/library/stats/help/lm.fit), etc, see below, for the actual numerical computations. For programming only, you may consider doing likewise.

All of weights, subset and offset are evaluated in the same way as variables in formula, that is first in data and then in the environment of formula.

**Value**

lm returns an object of [class](http://127.0.0.1:14695/library/stats/help/class) "lm" or for multiple responses of class c("mlm", "lm").

The functions summary and [anova](http://127.0.0.1:14695/library/stats/help/anova) are used to obtain and print a summary and analysis of variance table of the results. The generic accessor functions coefficients, effects, fitted.values and residuals extract various useful features of the value returned by lm.

An object of class "lm" is a list containing at least the following components:

|  |  |
| --- | --- |
| coefficients | a named vector of coefficients |
| residuals | the residuals, that is response minus fitted values. |
| fitted.values | the fitted mean values. |
| rank | the numeric rank of the fitted linear model. |
| weights | (only for weighted fits) the specified weights. |
| df.residual | the residual degrees of freedom. |
| call | the matched call. |
| terms | the [terms](http://127.0.0.1:14695/library/stats/help/terms) object used. |
| contrasts | (only where relevant) the contrasts used. |
| xlevels | (only where relevant) a record of the levels of the factors used in fitting. |
| offset | the offset used (missing if none were used). |
| y | if requested, the response used. |
| x | if requested, the model matrix used. |
| model | if requested (the default), the model frame used. |
| na.action | (where relevant) information returned by [model.frame](http://127.0.0.1:14695/library/stats/help/model.frame) on the special handling of NAs. |

In addition, non-null fits will have components assign, effects and (unless not requested) qr relating to the linear fit, for use by extractor functions such as summary and [effects](http://127.0.0.1:14695/library/stats/help/effects).

**Using time series**

Considerable care is needed when using lm with time series.

Unless na.action = NULL, the time series attributes are stripped from the variables before the regression is done. (This is necessary as omitting NAs would invalidate the time series attributes, and if NAs are omitted in the middle of the series the result would no longer be a regular time series.)

Even if the time series attributes are retained, they are not used to line up series, so that the time shift of a lagged or differenced regressor would be ignored. It is good practice to prepare a data argument by [ts.intersect](http://127.0.0.1:14695/library/stats/help/ts.intersect)(..., dframe = TRUE), then apply a suitable na.action to that data frame and call lm with na.action = NULL so that residuals and fitted values are time series.

**Note**

Offsets specified by offset will not be included in predictions by [predict.lm](http://127.0.0.1:14695/library/stats/help/predict.lm), whereas those specified by an offset term in the formula will be.

**Author(s)**

The design was inspired by the S function of the same name described in Chambers (1992). The implementation of model formula by Ross Ihaka was based on Wilkinson & Rogers (1973).

**References**

Chambers, J. M. (1992) *Linear models.* Chapter 4 of *Statistical Models in S* eds J. M. Chambers and T. J. Hastie, Wadsworth & Brooks/Cole.

Wilkinson, G. N. and Rogers, C. E. (1973) Symbolic descriptions of factorial models for analysis of variance. *Applied Statistics*, **22**, 392–9.

**See Also**

[summary.lm](http://127.0.0.1:14695/library/stats/help/summary.lm) for summaries and [anova.lm](http://127.0.0.1:14695/library/stats/help/anova.lm) for the ANOVA table; [aov](http://127.0.0.1:14695/library/stats/help/aov) for a different interface.

The generic functions [coef](http://127.0.0.1:14695/library/stats/help/coef), [effects](http://127.0.0.1:14695/library/stats/help/effects), [residuals](http://127.0.0.1:14695/library/stats/help/residuals), [fitted](http://127.0.0.1:14695/library/stats/help/fitted), [vcov](http://127.0.0.1:14695/library/stats/help/vcov).

[predict.lm](http://127.0.0.1:14695/library/stats/help/predict.lm) (via [predict](http://127.0.0.1:14695/library/stats/help/predict)) for prediction, including confidence and prediction intervals; [confint](http://127.0.0.1:14695/library/stats/help/confint) for confidence intervals of *parameters*.

[lm.influence](http://127.0.0.1:14695/library/stats/help/lm.influence) for regression diagnostics, and [glm](http://127.0.0.1:14695/library/stats/help/glm) for **generalized** linear models.

The underlying low level functions, [lm.fit](http://127.0.0.1:14695/library/stats/help/lm.fit) for plain, and [lm.wfit](http://127.0.0.1:14695/library/stats/help/lm.wfit) for weighted regression fitting.

More lm() examples are available e.g., in [anscombe](http://127.0.0.1:14695/library/stats/help/anscombe), [attitude](http://127.0.0.1:14695/library/stats/help/attitude), [freeny](http://127.0.0.1:14695/library/stats/help/freeny), [LifeCycleSavings](http://127.0.0.1:14695/library/stats/help/LifeCycleSavings), [longley](http://127.0.0.1:14695/library/stats/help/longley), [stackloss](http://127.0.0.1:14695/library/stats/help/stackloss), [swiss](http://127.0.0.1:14695/library/stats/help/swiss).

biglm in package [**biglm**](https://cran.r-project.org/package=biglm) for an alternative way to fit linear models to large datasets (especially those with many cases).

**Examples**

require(graphics)

## Annette Dobson (1990) "An Introduction to Generalized Linear Models".

## Page 9: Plant Weight Data.

ctl <- c(4.17,5.58,5.18,6.11,4.50,4.61,5.17,4.53,5.33,5.14)

trt <- c(4.81,4.17,4.41,3.59,5.87,3.83,6.03,4.89,4.32,4.69)

group <- gl(2, 10, 20, labels = c("Ctl","Trt"))

weight <- c(ctl, trt)

lm.D9 <- lm(weight ~ group)

lm.D90 <- lm(weight ~ group - 1) # omitting intercept

anova(lm.D9)

summary(lm.D90)

opar <- par(mfrow = c(2,2), oma = c(0, 0, 1.1, 0))

plot(lm.D9, las = 1) # Residuals, Fitted, ...

par(opar)

### less simple examples in "See Also" above