

# Package: glm4 (via r-universe)

May 18, 2026

**Type** Package

**Title** Fitting Generalized Linear Models Using Sparse Matrices

**Version** 0.1.0

**Description** Fits Generalised Linear Models (GLMs) with sparse and dense 'Matrix' matrices for memory efficiency. Acts as a wrapper for the glm4() function in the 'MatrixModels' package <doi:10.32614/CRAN.package.MatrixModels>, but adds convenient model methods and functions designed to mimic those associated with the glm() function from the 'stats' package.

**URL** <https://github.com/awhug/glm4/issues>

**BugReports** <https://github.com/awhug/glm4/issues>

**License** GPL (>= 2)

**Encoding** UTF-8

**RoxygenNote** 7.3.3

**Imports** methods, Matrix, MatrixModels, stats

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**Repository** <https://awhug.r-universe.dev>

**Date/Publication** 2026-04-18 00:13:14 UTC

**RemoteUrl** <https://github.com/awhug/glm4>

**RemoteRef** HEAD

**RemoteSha** 1a44c8e14151dff55c2947612aa4970c47423641

## Contents

|                        |   |
|------------------------|---|
| anova.glm4 . . . . .   | 2 |
| glm4 . . . . .         | 3 |
| summary.glm4 . . . . . | 4 |

|              |          |
|--------------|----------|
| <b>Index</b> | <b>6</b> |
|--------------|----------|

anova.glm4

*Analysis of Deviance for Generalized Linear Model Fits***Description**

Compute an analysis of deviance table for one or more generalized linear model fits.

**Usage**

```
## S3 method for class 'glm4'
anova(object, ..., dispersion = NULL, test = NULL)
```

**Arguments**

|            |   |
|------------|---|
| object     | an object of class "glm4"   |
| ...        | additional objects of class "glm4" for multi-model comparison   |
| dispersion | the dispersion parameter for the fitting family. By default it is obtained from the object(s).  |
| test       | a character string, (partially) matching one of "Chisq", "LRT", "Rao", "F" or "Cp". See <a href="#">stat.anova</a> . Or logical FALSE, which suppresses any test. |

**Details**

Specifying a single object gives a sequential analysis of deviance table for that fit. That is, the reductions in the residual deviance as each term of the formula is added in turn are given in as the rows of a table, plus the residual deviances themselves.

If more than one object is specified, the table has a row for the residual degrees of freedom and deviance for each model. For all but the first model, the change in degrees of freedom and deviance is also given. (This only makes statistical sense if the models are nested.) It is conventional to list the models from smallest to largest, but this is up to the user.

The table will optionally contain test statistics (and P values) comparing the reduction in deviance for the row to the residuals. For models with known dispersion (e.g., binomial and Poisson fits) the chi-squared test is most appropriate, and for those with dispersion estimated by moments (e.g., gaussian, quasibinomial and quasipoisson fits) the F test is most appropriate. If `anova.glm` can determine which of these cases applies then by default it will use one of the above tests. If the dispersion argument is supplied, the dispersion is considered known and the chi-squared test will be used. Argument `test=FALSE` suppresses the test statistics and P values. Mallows'  $C_p$  statistic is the residual deviance plus twice the estimate of  $\sigma^2$  times the residual degrees of freedom, which is closely related to AIC (and a multiple of it if the dispersion is known). You can also choose "LRT" and "Rao" for likelihood ratio tests and Rao's efficient score test. The former is synonymous with "Chisq" (although both have an asymptotic chi-square distribution).

The dispersion estimate will be taken from the largest model, using the value returned by [summary.glm](#). As this will in most cases use a Chi-squared-based estimate, the F tests are not based on the residual deviance in the analysis of deviance table shown.

**Value**

An object of class "anova" inheriting from class "data.frame".

**Warning**

The comparison between two or more models will only be valid if they are fitted to the same dataset. This may be a problem if there are missing values and R's default of `na.action = na.omit` is used, and `anova` will detect this with an error.

**References**

Hastie, T. J. and Pregibon, D. (1992) *Generalized linear models*. Chapter 6 of *Statistical Models in S* eds J. M. Chambers and T. J. Hastie, Wadsworth & Brooks/Cole.

**See Also**

[glm](#), [anova](#).

[drop1](#) for so-called 'type II' ANOVA where each term is dropped one at a time respecting their hierarchy.

**Examples**

```
## --- Continuing the Example from '?glm':  
  
anova(glm.D93, test = FALSE)  
anova(glm.D93, test = "Cp")  
anova(glm.D93, test = "Chisq")  
glm.D93a <-  
  update(glm.D93, ~treatment*outcome) # equivalent to Pearson Chi-square  
anova(glm.D93, glm.D93a, test = "Rao")
```

**Description**

'`glm4()`', is used to fit generalized linear models, specified by giving a symbolic description of the linear predictor and a description of the error distribution.

It is very similar to the standard '`stats::glm()`' function, but supports sparse matrices via the Matrix package, which can dramatically improve memory and computational efficiency on large and/or high-dimensional data.

**Usage**

```
glm4(formula, data, ...)
```

**Arguments**

|         |  |
|---------|--|
| formula | an object of class " <code>formula</code> " (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 <code>as.data.frame</code> to a 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>glm</code> is called. |
| ...     | potentially arguments passed on to fitter functions; not used currently.   |

**Details**

This function is a wrapper for `'MatrixModels::glm4()'` which returns a more user-friendly object designed to resemble `'stats::glm()'` as closely as possible. Behind the scenes, it extracts the relevant model details from the S4 class `'MatrixModels::glm4()'` object and calculates new ones where necessary (e.g. AIC, deviance, residual degrees of freedom, etc) as per `'stats::glm()'`.

Sparse matrix storage is not enabled by default; pass `'sparse = TRUE'` to use it.

**Value**

A list object of class `'glm4'`. See `'stats::glm()'` for more details on returned components.

**Examples**

```
fit <- glm4(mpg ~ cyl + wt, data = mtcars, family = gaussian())
print(fit)
```

---

summary.glm4

*Summarising Generalized Linear Models Using Sparse Matrices*


---

**Description**

Generates a summary of the `'glm4'` object to evaluate coefficients, standard errors, model fit criteria, etc.

**Usage**

```
## S3 method for class 'glm4'
summary(
  object,
  p.adjust = NULL,
  dispersion = NULL,
  correlation = FALSE,
  symbolic.cor = FALSE,
  ...
)
```

**Arguments**

|              |  |
|--------------|--|
| object       | an object of class "glm4"  |
| p.adjust     | returns p-values adjusted using one of several methods implemented in 'stats::p.adjust'. Defaults to 'NULL' for no adjustment, consistent with 'stats::glm'.               |
| dispersion   | the dispersion parameter for the family used. Either a single numerical value or NULL (the default), when it is inferred from object (see 'stats::summary.glm()' details). |
| correlation  | logical; if 'TRUE', the correlation matrix of the estimated parameters is returned and printed.  |
| symbolic.cor | logical; if 'TRUE', and 'correlation' is 'TRUE', the correlation matrix is printed in symbolic form.   |
| ...          | further arguments passed to or from other methods.   |

**Details**

This function is designed to resemble 'stats::summary.glm()' as closely as possible. It calculates the (un)scaled variance-covariance matrix from a 'MatrixModels::glm4()' object using 'Matrix::chol2inv()' and produces a coefficient table for easy inspection of model parameters.

**Value**

A list object of class 'c("summary.glm", "summary.glm4")'. See 'stats::summary.glm()' for more details on returned components.

**Examples**

```
fit <- glm4(mpg ~ cyl + wt, data = mtcars, family = gaussian())
summary(fit)
```

# Index

`anova`, 3  
`anova.glm4`, 2  
`as.data.frame`, 4  
  
`drop1`, 3  
  
`formula`, 4  
  
`glm`, 3  
`glm4`, 3  
  
`stat.anova`, 2  
`summary.glm`, 2  
`summary.glm4`, 4