

Package: glm2 (via r-universe)

October 10, 2026

Type Package

Title Fitting Generalized Linear Models

Description Fits generalized linear models using the same model specification as glm in the stats package, but with a modified default fitting method that provides greater stability for models that may fail to converge using glm.

Version 1.2.1.9000

Depends R (>= 3.2.0), stats

Suggests Matrix, methods

License GPL (>=2)

LazyData true

URL <https://github.com/mdonoghoe/glm2>

Date 2025-06-20

RoxygenNote 7.3.2

Encoding UTF-8

Repository <https://mdonoghoe.r-universe.dev>

Date/Publication 2025-06-20 01:05:57 UTC

RemoteUrl <https://github.com/mdonoghoe/glm2>

RemoteRef HEAD

RemoteSha e666fa064e6e89cc6871c50e68a7f483a1a7c3f3

Contents

glm2-package	2
crabs	3
dfbetas.glm2Matrix	3
glm.fit2	4
glm2	6
heart	10
influence.glm2Matrix	11

predict.glm2Matrix	11
summary.glm2Matrix	12
vcov.glm2Matrix	12

Index	13
--------------	-----------

glm2-package	<i>Fitting Generalized Linear Models</i>
--------------	--

Description

Fits generalized linear models using the same model specification as `glm` in the **stats** package, but with a modified default fitting method. The method provides greater stability for models that may fail to converge using `glm`.

Details

Package: glm2
Type: Package
Version: 1.2.1
License: GPL (>=2)
LazyData: true

There are two functions in the package, `glm2` and `glm.fit2`. The `glm2` function fits generalized linear models using the same model specification as `glm` in the **stats** package. It is identical to `glm` except for minor modifications to change the default fitting method. The `glm.fit2` function provides the default fitting method for `glm2`. It is identical to `glm.fit` in the **stats** package, except for modifications to the computational method that provide more stable convergence. Normally only `glm2` would be called directly, although like `glm.fit`, `glm.fit2` can be called directly. It can also be passed to `glm` as an alternative fitting method, using the `method` argument. See Marschner (2011) for a discussion of the need for a modified fitting method.

Author(s)

Ian Marschner (using code from `glm` and `glm.fit` in the **stats** package)
Maintainer: Mark W. Donoghoe <markdonoghoe@gmail.com>

References

Marschner, I.C. (2011) `glm2`: Fitting generalized linear models with convergence problems. The R Journal, Vol. 3/2, pp.12-15.

See Also

- Useful links:
- <https://github.com/mdonoghoe/glm2>

crabs

*Horseshoe Crab Data***Description**

This data set is derived from Agresti (2007, Table 3.2, pp.76-77). It gives 4 variables for each of 173 female horseshoe crabs. Also provided are two random samples of the data with replacement, which are useful for illustrating the convergence properties of `glm` and `glm2`.

Format

A data frame with 173 observations on the following 6 variables:

Satellites number of male partners in addition to the female's primary partner

Width width of the female in centimeters

Dark a binary factor indicating whether the female has dark coloring (yes or no)

GoodSpine a binary factor indicating whether the female has good spine condition (yes or no)

Rep1 a random sample with replacement from 1:173

Rep2 a second random sample with replacement from 1:173

Details

The variables `Dark` and `GoodSpine` are derived from the raw data. In the notation of Table 3.2 of Agresti (2007), `Dark = yes` corresponds to $C > 3$ and `GoodSpine = yes` corresponds to $S < 3$. The two random samples `Rep1` and `Rep2` can be used to provide random samples with replacement from the full data set. These two random samples are useful for illustrating the convergence properties of `glm` and `glm2`; see examples in the help documentation for `glm2`.

References

Agresti, A. (2007) An Introduction to Categorical Data Analysis (2nd ed.). Hoboken, NJ: Wiley.

dfbetas.glm2Matrix

*Regression Deletion Diagnostics (not yet implemented)***Description**

This method is a placeholder. The `dfbetas` method for GLMs fit using `glm2(..., method = "glm.fit2.Matrix")` is not yet implemented.

Value

An error indicating the method is not implemented.

Description

glm.fit2 is identical to glm.fit in the **stats** package, except for a modification to the computational method that provides improved convergence properties. It is the default fitting method for glm2 and can also be used as an alternative fitting method for glm, instead of the default method glm.fit. glm.fit2.Matrix is similar, but allows for the model matrix x to be supplied as a sparse [Matrix](#), allowing for greater efficiency with large data.

Usage

```
glm.fit2(  
  x,  
  y,  
  weights = rep(1, nobs),  
  start = NULL,  
  etastart = NULL,  
  mustart = NULL,  
  offset = rep(0, nobs),  
  family = gaussian(),  
  control = list(),  
  intercept = TRUE,  
  singular.ok = TRUE  
)
```

```
glm.fit2.Matrix(  
  x,  
  y,  
  weights = rep(1, nobs),  
  start = NULL,  
  etastart = NULL,  
  mustart = NULL,  
  offset = rep(0, nobs),  
  family = gaussian(),  
  control = list(),  
  intercept = TRUE,  
  singular.ok = TRUE  
)
```

Arguments

x	as for glm.fit ; a sparse Matrix for glm.fit2.Matrix
y	as for glm.fit
weights	as for glm.fit

start	as for glm.fit
etastart	as for glm.fit
mustart	as for glm.fit
offset	as for glm.fit
family	as for glm.fit
control	as for glm.fit
intercept	as for glm.fit
singular.ok	as for glm.fit . For glm.fit2.Matrix it is ideal if you first ensure there are no linear dependencies and use singular.ok = FALSE

Details

glm.fit2 is a modified version of glm.fit in the **stats** package. The computational method in glm.fit2 uses a stricter form of step-halving to deal with numerical instability in the iteratively reweighted least squares algorithm. Whereas glm.fit uses step-halving to correct divergence and parameter space violations, glm.fit2 additionally uses step-halving to force the model deviance to decrease at each iteration, which improves the convergence properties. Like glm.fit, glm.fit2 usually would not be called directly. Instead, it is called by glm2 as the default fitting method. Like glm.fit, it is possible to call glm.fit2 directly if the response vector and design matrix have already been calculated, in which case it may be more efficient than calling glm2. It can also be passed to glm in the **stats** package, using the method argument, providing an alternative to the default fitting method glm.fit. See Marschner (2011) for a discussion of the need for a modified fitting method.

glm.fit2.Matrix uses the same algorithm as glm.fit2, but employs [QR Factorization methods](#) from the **Matrix** package, which are more computationally efficient with large, sparse model matrices. When calling via [glm2](#), this is generated using `Matrix::sparse.model.matrix`. If you are calling glm.fit2.Matrix directly, ideally you should supply a Matrix object. This approach will not be as efficient if called using `glm(..., method = "glm.fit2.Matrix")`.

Value

The value returned by glm.fit2 has exactly the same documentation as the value returned by glm.fit. The value returned by glm.fit2.Matrix is slightly different in some details.

Author(s)

glm.fit2 uses the code from glm.fit, whose authors are listed in the help documentation for the **stats** package. Modifications to this code were made by Ian Marschner.

References

Marschner, I.C. (2011) glm2: Fitting generalized linear models with convergence problems. The R Journal, Vol. 3/2, pp.12-15.

See Also

[glm.fit](#)

Examples

```
library(glm2)

#--- logistic regression null model -----#
# (intercept estimate = log(0.75/0.25) = 1.098612)

y <- c(1,1,1,0)
x <- rep(1,4)

#--- divergence of glm.fit to infinite estimate ---#
fit1 <- glm.fit(x,y, family=binomial(link="logit"),start=-1.81)
fit2 <- glm.fit2(x,y, family=binomial(link="logit"),start=-1.81)
print.noquote(c(fit1$coef,fit2$coef))
```

glm2

Fitting Generalized Linear Models

Description

Fits generalized linear models using the same model specification as `glm` in the **stats** package, but with a modified default fitting method. The method provides greater stability for models that may fail to converge using `glm`.

Usage

```
glm2(
  formula,
  family = gaussian,
  data,
  weights,
  subset,
  na.action,
  start = NULL,
  etastart,
  mustart,
  offset,
  control = list(...),
  model = TRUE,
  method = "glm.fit2",
  x = FALSE,
  y = TRUE,
  singular.ok = TRUE,
  contrasts = NULL,
  ...
)
```

Arguments

formula	as for glm
family	as for glm
data	as for glm
weights	as for glm
subset	as for glm
na.action	as for glm
start	as for glm
etastart	as for glm
mustart	as for glm
offset	as for glm
control	as for glm except by default control is passed to <code>glm.fit2</code> instead of <code>glm.fit</code>
model	as for glm
method	the method used in fitting the model. The default method " <code>glm.fit2</code> " uses iteratively reweighted least squares with modified step-halving that forces the deviance to decrease at each iteration; see help documentation for glm.fit2 . method = " <code>glm.fit2.Matrix</code> " uses the same algorithm but may be more efficient for large, sparse models by employing the Matrix package. As in <code>glm</code> , the alternative method " <code>model.frame</code> " returns the model frame and does no fitting.
x	as for glm
y	as for glm
singular.ok	as for glm . NB this is ignored (and defaults to TRUE) for R versions < 3.5.0.
contrasts	as for glm
...	as for glm

Details

`glm2` is a modified version of `glm` in the **stats** package. It fits generalized linear models using the same model specification as `glm`. It is identical to `glm` except for minor modifications to change the default fitting method. The default method uses a stricter form of step-halving to force the deviance to decrease at each iteration and is implemented in `glm.fit2`. Like `glm`, user-supplied fitting functions can be used with `glm2` by passing a function or a character string naming a function to the `method` argument. See Marschner (2011) for a discussion of the need for a modified fitting method.

Value

The value returned by `glm2` has exactly the same documentation as the value returned by `glm`, except for:

method	the name of the fitter function used, which by default is " <code>glm.fit2</code> ".
--------	--

Author(s)

glm2 uses the code from glm, whose authors are listed in the help documentation for the **stats** package. Modifications to this code were made by Ian Marschner.

References

Marschner, I.C. (2011) glm2: Fitting generalized linear models with convergence problems. The R Journal, Vol. 3/2, pp.12-15.

See Also

[glm](#)

Examples

```
library(glm2)
data(crabs)
data(heart)

#####
# EXAMPLE 1: logistic regression null model
# (behaviour of glm and glm2 for different starting values)
#####

y <- c(1,1,1,0)
# intercept estimate = log(0.75/0.25) = 1.098612

#--- identical behaviour ---#
fit1 <- glm(y ~ 1, family=binomial(link="logit"),
  control=glm.control(trace=TRUE))
fit2 <- glm2(y ~ 1, family=binomial(link="logit"),
  control=glm.control(trace=TRUE))
print.noquote(c(fit1$coef,fit2$coef))

#--- convergence via different paths ---#
fit1 <- glm(y ~ 1, family=binomial(link="logit"),start=-1.75,
  control=glm.control(trace=TRUE))
fit2 <- glm2(y ~ 1, family=binomial(link="logit"),start=-1.75,
  control=glm.control(trace=TRUE))
print.noquote(c(fit1$coef,fit2$coef))

#--- divergence of glm to infinite estimate ---#
fit1 <- glm(y ~ 1, family=binomial(link="logit"),start=-1.81)
fit2 <- glm2(y ~ 1, family=binomial(link="logit"),start=-1.81)
print.noquote(c(fit1$coef,fit2$coef))

#####
# EXAMPLE 2: identity link Poisson (successful boundary convergence
# using 4 identical approaches in glm and glm2 with the method argument)
#####
```

```

satellites <- crabs$Satellites
width.shifted <- crabs$Width - min(crabs$Width)
dark <- crabs$Dark
goodspine <- crabs$GoodSpine

fit1 <- glm(satellites ~ width.shifted + factor(dark) + factor(goodspine),
  family = poisson(link="identity"), start = rep(1,4))

fit2 <- glm2(satellites ~ width.shifted + factor(dark) + factor(goodspine),
  family = poisson(link="identity"), start = rep(1,4))

fit1.eq <- glm2(satellites ~ width.shifted + factor(dark) + factor(goodspine),
  family = poisson(link="identity"), start = rep(1,4), method = "glm.fit")

fit2.eq <- glm(satellites ~ width.shifted + factor(dark) + factor(goodspine),
  family = poisson(link="identity"), start = rep(1,4), method = "glm.fit2")

noquote(c("deviances: ", fit1$dev, fit2$dev, fit1.eq$dev, fit2.eq$dev))
noquote(c("converged: ", fit1$conv, fit2$conv, fit1.eq$conv, fit2.eq$conv))
noquote(c("boundary: ", fit1$bound, fit2$bound, fit1.eq$bound, fit2.eq$bound))

#=====
# EXAMPLE 3: identity link Poisson (periodic non-convergence in glm)
#=====

R1 <- crabs$Rep1
satellites <- crabs$Satellites[R1]
width.shifted <- crabs$Width[R1] - min(crabs$Width)
dark <- crabs$Dark[R1]
goodspine <- crabs$GoodSpine[R1]

fit1 <- glm(satellites ~ width.shifted + factor(dark) + factor(goodspine),
  family = poisson(link="identity"), start = rep(1,4),
  control = glm.control(trace=TRUE))

fit2 <- glm2(satellites ~ width.shifted + factor(dark) + factor(goodspine),
  family = poisson(link="identity"), start = rep(1,4),
  control = glm.control(trace=TRUE))

noquote(c("deviances: ", fit1$dev, fit2$dev))
noquote(c("converged: ", fit1$conv, fit2$conv))

#=====
# EXAMPLE 4: log link binomial (periodic non-convergence in glm)
#=====

patients <- heart$Patients
deaths <- heart$Deaths
agegroup <- heart$AgeGroup
severity <- heart$Severity
delay <- heart$Delay
region <- heart$Region
start.p <- sum(deaths)/sum(patients)

```

```

fit1 <- glm(cbind(deaths,patients-deaths) ~ factor(agegroup) + factor(severity)
+ factor(delay) + factor(region), family = binomial(link="log"),
start = c(log(start.p), rep(0,8)), control = glm.control(trace=TRUE,maxit=100))

fit2 <- glm2(cbind(deaths,patients-deaths) ~ factor(agegroup) + factor(severity)
+ factor(delay) + factor(region), family = binomial(link="log"),
start = c(log(start.p), rep(0,8)), control = glm.control(trace=TRUE))

noquote(c("deviances: ",fit1$dev,fit2$dev))
noquote(c("converged: ",fit1$conv,fit2$conv))

#=====
# EXAMPLE 5: identity link Poisson (aperiodic non-convergence in glm)
#=====

R2 <- crabs$Rep2
satellites <- crabs$Satellites[R2]
width.shifted <- crabs$Width[R2] - min(crabs$Width)
dark <- crabs$Dark[R2]
goodspine <- crabs$GoodSpine[R2]

fit1 <- glm(satellites ~ width.shifted + factor(dark) + factor(goodspine),
family = poisson(link="identity"), start = rep(1,4),
control = glm.control(trace=TRUE))

fit2 <- glm2(satellites ~ width.shifted + factor(dark) + factor(goodspine),
family = poisson(link="identity"), start = rep(1,4),
control = glm.control(trace=TRUE))

noquote(c("deviances: ",fit1$dev,fit2$dev))
noquote(c("converged: ",fit1$conv,fit2$conv))

```

heart

Heart Attack Data

Description

This data set is a cross-tabulation of data on 16949 individuals who experienced a heart attack (ASSENT-2 Investigators, 1999). There are 4 categorical factors each at 3 levels, together with the number of patients and the number of deaths for each observed combination of the factors. This data set is useful for illustrating the convergence properties of `glm` and `glm2`.

Format

A data frame with 74 observations on the following 6 variables.

Deaths number of deaths

Patients number of patients

AgeGroup categorization of the age of the patients

Severity severity of the heart attack

Delay categorization of the time from heart attack to treatment

Region geographical region in which the patients were treated

Details

The variable AgeGroup groups the age of the patients as follows: 1 = <65 years, 2 = 65-75 years, 3 = >75 years. The variable Delay groups the time from heart attack onset to treatment as follows: 1 = <2 hours, 2 = 2-4 hours, 3 = >4 hours. The variable Severity denotes the severity of the heart attack using the Killip class: 1 = least severe (class I), 2 = middle severity (class II), 3 = most severe (class III or IV). The variable Region provides the geographical region in which the patients were treated: 1 = Western countries, 2 = Latin America, 3 = Eastern Europe. This data set is useful for illustrating the convergence properties of glm and glm2; see examples in the help documentation for glm2.

References

ASSENT-2 Investigators. (1999) Single-bolus tenecteplase compared with front-loaded alteplase in acute myocardial infarction: The ASSENT-2 double-blind randomized trial. *Lancet* 354, 716-722.

influence.glm2Matrix *Regression Diagnostics (not yet implemented)*

Description

This method is a placeholder. The `influence` method for GLMs fit using `glm2(..., method = "glm.fit2.Matrix")` is not yet implemented.

Value

An error indicating the method is not implemented.

predict.glm2Matrix *Predict Method for Fits*

Description

Implementation of `predict.glm` for GLMs fit using `glm2(..., method = "glm.fit2.Matrix")`. See its documentation for full details.

summary.glm2Matrix	<i>Summarizing Generalized Linear Model Fits</i>
--------------------	--

Description

Implementation of [summary.glm](#) for GLMs fit using `glm2(..., method = "glm.fit2.Matrix")`. See its documentation for full details.

vcov.glm2Matrix	<i>Calculate Variance-Covariance Matrix for a Fitted Model Object</i>
-----------------	---

Description

Implementation of [vcov](#) for GLMs fit using `glm2(..., method = "glm.fit2.Matrix")`. See its documentation for full details.

Index

- * **datasets**
 - crabs, [3](#)
 - heart, [10](#)
- * **package**
 - glm2-package, [2](#)
- crabs, [3](#)
- dfbetas, [3](#)
- dfbetas.glm2Matrix, [3](#)
- glm, [7](#), [8](#)
- glm.fit, [4](#), [5](#)
- glm.fit2, [4](#), [7](#)
- glm.fit2.Matrix, [3](#), [7](#), [11](#), [12](#)
- glm2, [5](#), [6](#)
- glm2-package, [2](#)
- heart, [10](#)
- influence, [11](#)
- influence.glm2Matrix, [11](#)
- Matrix, [4](#)
- Matrix::sparse.model.matrix, [5](#)
- predict.glm, [11](#)
- predict.glm2Matrix, [11](#)
- QR Factorization methods, [5](#)
- summary.glm, [12](#)
- summary.glm2Matrix, [12](#)
- vcov, [12](#)
- vcov.glm2Matrix, [12](#)
- vcov.summary.glm2Matrix
 (vcov.glm2Matrix), [12](#)