Package ‘eflm’

May 31, 2021

**Title**  Efficient Fitting of Linear and Generalized Linear Models

**Version**  0.3.0

**Description**  Efficient Fitting of Linear and Generalized Linear Models by using just base R. As an alternative to lm() and glm(), this package provides elm() and eglm(), with a significant speedup when the number of observations is larger than the number of parameters to estimate. The speed gains are obtained by reducing the NxP model matrix to a PxP matrix, and the best computational performance is obtained when R is linked against 'OpenBLAS', 'Intel MKL' or other optimized 'BLAS' library. This implementation aims at being compatible with 'broom' and 'sandwich' packages for summary statistics and clustering by providing S3 methods.

**URL**  https://github.com/pachadotdev/eflm/

**BugReports**  https://github.com/pachadotdev/eflm/issues/

**License**  Apache License (>= 2)

**Imports**  stats, tibble

**Suggests**  broom, sandwich, testthat, patrick, rlang, covr

**Encoding**  UTF-8

**Depends**  R (>= 4.0)

**RoxygenNote**  7.1.1

**NeedsCompilation**  no

**Author**  Mauricio Vargas [aut, cre] (https://orcid.org/0000-0003-1017-7574),
            Alexey Kravchenko [ths],
            Constanza Prado [ths],
            Yoto Yotov [ths],
            Neal Fulz [rev] (independent code auditing),
            Rahul Saxena [ctb] (improved broom integration),
            The United Nations [dct, cph, fnd]

**Maintainer**  Mauricio Vargas <mavargas11@uc.cl>

**Repository**  CRAN

**Date/Publication**  2021-05-31 21:20:02 UTC
Description

Efficient Generalized Linear Model ("eglm") is used to fit generalized linear models in an equivalent way to "glm" but in a reduced time depending on the design matrix and the family (or link).

Usage

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

Arguments

formula an object of class "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'.

family

a description of the error distribution and link function to be used in the model.
This can be a character string naming a family function, a family function or the
result of a call to a family function. See family for details of family functions.

data

an optional data frame, list or environment (or object coercible by 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.

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.

subset

an optional vector specifying a subset of observations to be used in the fitting
process.

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, and is na.fail if that is
unset. The ‘factory-fresh’ default is na.omit. Another possible value is NULL,
no action. Value na.exclude can be useful.

start

starting values for the parameters in the linear predictor.

etastart

starting values for the linear predictor.

mustart

starting values for the vector of means.

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 or matrix
of extents matching those of the response. One or more 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.

control

a list of parameters for controlling the fitting process. For eglm.wfit this is
passed to glm.control.

model

a logical value indicating whether model frame should be included as a compo-
nent of the returned value.

method

the method to be used in fitting the model. The default method "eglm.wfit"
uses iteratively reweighted least squares (IWLS); the alternative "model.frame"
returns the model frame and does no fitting. User-supplied fitting functions can
be supplied either as a function or a character string naming a function, with a
function which takes the same arguments as glm.fit from the stats package. If
specified as a character string it is looked up from within the eglm namespace.

x, y

logical values indicating whether the model matrix (x) and the response vector
(y) used in the fitting process should be returned as components of the returned
value.

singular.ok

logical; if FALSE a singular fit is an error.

contrasts

an optional list. See the contrasts.arg of model.matrix.default.

reduce

logical; if TRUE an alternate design matrix of p * p is used for the fitting instead
of the traditional n * p design matrix.

... For eglm: arguments to be used to form the default control argument if it is
not supplied directly. For weights: further arguments passed to or from other
methods.
Details

Models for eglm 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, and exactly the same as "glm" from the stats package.

Value

An object of class "eglm" that behaves the same way as the "glm" class, see the function "glm". This output also includes the logical "reduce" and, depending on it, the reduced design matrix "xtx" when the reduce argument is set to TRUE.

Examples

eglm(mpg ~ wt, family = gaussian, data = mtcars)
Arguments

- **x, y**
  - For eglm.wfit: x is a design matrix of dimension \( n \times p \), and y is a vector of observations of length n, or a matrix with n rows.

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 \( \text{sum}(w*e^2) \)); otherwise ordinary least squares is used.

start
- starting values for the parameters in the linear predictor.

etastart
- starting values for the linear predictor.

mustart
- starting values for the vector of means.

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 or matrix of extents matching those of the response. One or more 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.

family
- a description of the error distribution and link function to be used in the model. This can be a character string naming a family function, a family function or the result of a call to a family function. See family for details of family functions.

control
- a list of parameters for controlling the fitting process. For eglm.wfit this is passed to glm.control.

intercept
- logical value indicating whether intercept should be included in the null model. Defaults to TRUE.

singular.ok
- logical; if FALSE a singular fit is an error.

reduce
- logical; if TRUE an alternate design matrix of \( p \times p \) is used for the fitting instead of the traditional \( n \times p \) design matrix.

Details

eglm.wfit is a workhorse function: it is not normally called directly but can be more efficient where the response vector, design matrix and family have already been calculated. Use eglm for most of the cases.

Value

A list that contains the same elements as the output from "glm.fit", with the addition of the vector "good" that indicates with logicals which observations were used in the fitting process.

Examples

```r
x <- cbind(rep(1, nrow(mtcars)), mtcars$wt)
y <- mtcars$mpg
eglm.wfit(x, y)
```
Description

Efficient Linear Model ("elm") is used to fit linear models in an equivalent way to "lm" but in a reduced time depending on the design matrix.

Usage

```r
elm(
  formula,
  data,
  subset,
  weights,
  na.action,
  method = "qr",
  model = TRUE,
  x = FALSE,
  y = FALSE,
  qr = TRUE,
  singular.ok = TRUE,
  contrasts = NULL,
  offset,
  reduce = TRUE,
  ...
)
```

Arguments

- **formula**: an object of class "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` 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.
- **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`, and is `na.fail` if that is unset. The ‘factory-fresh’ default is `na.omit`. Another possible value is `NULL`, no action. Value `na.exclude` can be useful.
the method to be used in fitting the model. The default method "eglm.wfit" uses iteratively reweighted least squares (IWLS); the alternative "model.frame" returns the model frame and does no fitting. User-supplied fitting functions can be supplied either as a function or a character string naming a function, with a function which takes the same arguments as glm.fit from the stats package. If specified as a character string it is looked up from within the eglm namespace.

a logical value indicating whether model frame should be included as a component of the returned value.

logical values indicating whether the model matrix \(x\) and the response vector \(y\) used in the fitting process should be returned as components of the returned value.

logical values indicating whether the model matrix \(x\) and the response vector \(y\) used in the fitting process should be returned as components of the returned value.

logical. If TRUE the corresponding QR decomposition component of the fit is returned.

logical; if FALSE a singular fit is an error.

an optional list. See the contrasts.arg of model.matrix.default.

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 or matrix of extents matching those of the response. One or more 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.

logical; if TRUE an alternate design matrix of \(p \times p\) is used for the fitting instead of the traditional \(n \times p\) design matrix.

For eglm: arguments to be used to form the default control argument if it is not supplied directly. For weights: further arguments passed to or from other methods.

Models for elm 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, and exactly the same as "lm" from the stats package.

An object of class "elm" that behaves the same way as the "lm" class, see the function "lm". This output also includes the logical "reduce" and, depending on it, the reduced design matrix "xtx" when the reduce argument is set to TRUE.
Examples

elm(mpg ~ wt, data = mtcars)

Description

Efficient Linear Model Weighted Fit ("elm.wfit") is used to fit linear models in an equivalent way to "glm.fit" but in a reduced time depending on the design matrix and the family (or link).

Usage

```r
elm.wfit(
  x, y,
  weights = rep.int(1, n),
  offset = NULL,
  method = "qr",
  tol = 1e-07,
  singular.ok = TRUE,
  reduce = TRUE,
  ...
)
```

Arguments

- `x` logical values indicating whether the model matrix (`x`) and the response vector (`y`) used in the fitting process should be returned as components of the returned value.
- `y` logical values indicating whether the model matrix (`x`) and the response vector (`y`) used in the fitting process should be returned as components of the returned value.
- `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(weights * e^2)`); otherwise ordinary least squares is used.
- `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 or matrix of extents matching those of the response. One or more `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`.
- `method` the method to be used in fitting the model. The default method "eglm.wfit" uses iteratively reweighted least squares (IWLS): the alternative "model.frame" returns the model frame and does no fitting. User-supplied fitting functions can...
be supplied either as a function or a character string naming a function, with a function which takes the same arguments as glm.fit from the stats package. If specified as a character string it is looked up from within the eflm namespace.

tol
tolerance for the qr decomposition. Default is 1e-7.
singular.ok
logical; if FALSE a singular fit is an error.
reduce
logical; if TRUE an alternate design matrix of p * p is used for the fitting instead of the traditional n * p design matrix.

For eglm: arguments to be used to form the default control argument if it is not supplied directly. For weights: further arguments passed to or from other methods.

Details
elm.wfit is a workhorse function: it is not normally called directly but can be more efficient where the response vector, design matrix and family have already been calculated. Use elm for most of the cases.

Value
A list that contains the same elements as the output from "lm.fit".

Examples
x <- cbind(rep(1, nrow(mtcars)), mtcars$wt)
y <- mtcars$mpg
elm.wfit(x, y)
Index

as.data.frame, 3, 6

eglm, 2
eglm.wfit, 4
elm, 6
elm.wfit, 8

family, 3, 5
formula, 2, 6

glm, 2, 4
glm.fit, 4, 5, 8

lm, 6, 7
lm.fit, 9

model.offset, 3, 5, 7, 8

na.exclude, 3, 6
na.fail, 3, 6
na.omit, 3, 6

offset, 3, 5, 7, 8
options, 3, 6

qr, 9

stats, 4, 7