Package 'TriLLIEM'

May 7, 2026

Type Package

Title Log-Linear Modelling of Triad Genotype Data

Version 0.1.1

Description Triad Log-Linear modelling of Imprinting
Environmental interactions, and Maternal effects (TriLLIEM).
This is an implementation of the log-linear model described in a series of
papers, see for example Ainsworth et al. (2010) <doi:10.1002/gepi.20547>.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports dplyr, rlang, Rdpack

Suggests knitr, rmarkdown, withr, testthat (>= 3.0.0)

RoxygenNote 7.3.1

Depends R (>= 4.1.0)

VignetteBuilder knitr

Config/testthat/edition 3

URL https://github.com/KevinHZhao/TriLLIEM

BugReports https://github.com/KevinHZhao/TriLLIEM/issues

RdMacros Rdpack

NeedsCompilation no

Author Kevin Heda Zhao [aut, cre],
Kelly Burkett [aut]

Maintainer Kevin Heda Zhao <trilliemmaintainer@gmail.com>

Repository CRAN

Date/Publication 2026-03-12 08:20:33 UTC

Contents

`anova.TriLLIEM` *ANOVA method for TriLLIEM objects*

Description

This method is modelled after the anova.glm method. It produces an analysis of deviance table for multiple nested models.

Usage

```
## S3 method for class 'TriLLIEM'
anova(object, ...)
```

Arguments

`object`	An object of class `TriLLIEM`.
`...`	Additional `TriLLIEM` objects for comparison to `object`

Details

Like anova.glm, each model's residual degrees of freedom and deviances are given, alongside their respective differences between the models. Models should be nested for these results to be statistically interpretable. The last column shows the p-value from chi-squared tests comparing the difference in deviance for each model.

`TriLLIEM` objects modelling any sort of imprinting effect must use this function, as the EM algorithm used in `TriLLIEM` causes the anova.glm function to treat the estimated values as having been truly observed, modifying the degrees of freedom.

Value

An object of class "`anova.TriLLIEM`" inheriting from class "`anova`".

See Also

anova.glm

Examples

```
model1 <- TriLLIEM(dat = example_dat4R, effects = c("C"))
model2 <- TriLLIEM(dat = example_dat4R, effects = c("C", "M"))
anova(model1, model2)
```

`example_dat4R` *Example triad data*

Description

Example data formatted to be used in the TriLLIEM function.

Usage

```
example_dat4R
```

Format

A dataframe with columns for:

type index for the category corresponding to maternal (M), paternal (F), and child (C) genotypes

mt_MS mating type category for maternal (M) and paternal (F) genotypes under a mating symmetry model

mt_MaS mating type category for maternal (M) and paternal (F) genotypes under a model that does not assume mating symmetry

M mother genotype

F father genotype

C child genotype

E binary variable for if environmental effects are present (1) or not present (0)

D disease present (1)/not present (0)

count counts for each category of maternal (M), paternal (F), and child (C) genotypes

`fullview` *View underlying simulation details for* `TriLLIEM.sim` *objects*

Description

View underlying simulation details for `TriLLIEM.sim` objects

Usage

```
fullview(sim)
```

Arguments

`sim`	an object of class `TriLLIEM.sim`

Value

A data frame with the same counts in each category as `sim`, but with underlying simulation probabilities given as columns:

"typeOrig" Index for each genetically distinct row, based on "M", "F", "C", "matOrg", and "patOrg". Ranges from 1 to 64, unlike the `type` column in `sim` which ranges from 1 to 60, as the inclusion of "matOrg" and "patOrg" lead to potentially four additional rows.

"matOrg" and "patOrg" Binary variables for when the minor allele is maternally/paternally inherited, so that counts when "M", "F", "C" are all 1 are not ambiguous when determining parent-of-origin effects.

"prMF" Probability of the mother ("M") and father ("F") pairing in the simulated population, based on the minor allele frequency and mating type coefficients given during simulation.

"prCGivenMFOrg" Probability of the child's genotype ("C") and the allele parent of origin ("matOrg" and "patOrg") conditional on the genotypes of the mother and father.

"prMFCOrg" Probability of the triad based on genotypes and parent of origin, product of "prMF" and "prCGivenMFOrg".

"PrMFCOrg" Probability of the triad based on genotypes, parent of origin, and environmental exposure ("E") conditional on disease status ("D"), obtained by scaling "prMFCOrg" by the risk of disease (equal to "prMFCOrg" when "D" is 0).

"pop" Sub-population each row is simulated from (all 1 if simulated without population stratification)

The "count" column in "sim" is the sum of the rows of the "count" column in the returned data frame when grouped by "M", "F", "C", "E", "D".

Examples

```
## View the underlying distributions behind some of the example models given in simulateData()
dat1 <- simulateData(S = c(1, 2, 4), If = 3)
fullview(dat1)

dat2 <- simulateData(
    nControl = 1000,
    propE = c(0.1, 0.4),
    propE.control = c(0.2, 0.2),
    nPop = 2,
    maf = c(0.3, 0.4),
    prev.byPop = c(0.2, 0.3),
    prop.byPop = c(0.6, 0.4)
 )
fullview(dat2)
```

---

print.summary.TriLLIEM

Print method for summary.TriLLIEM *objects*

Description

Print method for summary.TriLLIEM objects

Usage

```
## S3 method for class 'summary.TriLLIEM'
print(
  x,
  digits = max(3L, getOption("digits") - 3L),
  signif.stars = getOption("show.signif.stars"),
  ...
)
```

Arguments

x	an object of class "summary.TriLLIEM", usually, a result of a call to summary.TriLLIEM.
digits	the number of significant digits to use when printing.
signif.stars	logical. If TRUE, 'significance stars' are printed for each coefficient, with significance codes shown under the coefficients table.
...	additional arguments passed to printCoefmat.

Value

Prints summary of the TriLLIEM model, displaying the original call to the function, the matrix of coefficients, the AIC, and the number of Fisher Scoring and EM iterations.

See Also

print.summary.glm

Examples

```
res <- TriLLIEM(mtmodel = "HWE", effects = c("C", "M", "Im"), dat = example_dat4R)
print(summary(res))
```

`print.TriLLIEM` *Print method for* `TriLLIEM` *objects*

Description

Print method for `TriLLIEM` objects

Usage

```
## S3 method for class 'TriLLIEM'
print(x, digits = max(3L, getOption("digits") - 3L), ...)
```

Arguments

`x`	an object of class `TriLLIEM`, usually, a result of a call to TriLLIEM.
`digits`	the number of significant digits to use when printing.
`...`	arguments passed to or from other methods.

Value

Prints details of the `TriLLIEM` model, with nuisance parameters (e.g., mating type parameters) omitted. To view all fitted parameters, run `coef(x)`.

Examples

```
res <- TriLLIEM(mtmodel = "HWE", effects = c("C", "M", "Im"), dat = example_dat4R)
print(res)
```

`simulateData` *Simulate data for TriLLIEM*

Description

Function to simulate maternal, paternal, and child genotype counts under different genetic effect models.

Usage

```
simulateData(
  nCases = 1000,
  nControl = 0,
  R = c(1, 1, 1),
  S = c(1, 1, 1),
  V = c(1, 1, 1),
  mtCoef = c(1, 1, 1),
  Im = 1,
  If = 1,
  propE = 0,
  propE.control = propE,
  Einteraction = "M",
  nPop = 1,
  maf = 0.3,
  prev.byPop = NULL,
  prop.byPop = NULL,
  Fst = 0.005
)
```

Arguments

`nCases`	number of case trios to simulate
`nControl`	number of control trios
`R`	vector of 3 elements representing child effects for 0, 1, and 2 copies of the risk allele, respectively
`S`	vector of 3 elements representing maternal effects for 0, 1, and 2 copies of the risk allele, respectively
`V`	vector of 3 elements representing gene-environment effects for 0, 1, and 2 copies of the risk allele, respectively
`mtCoef`	vector of 3 elements representing the mating type coefficients, see details for more information
`Im`	maternal imprinting effect
`If`	paternal imprinting effect
`propE`	scalar or vector of proportion of case trios in the environmental exposure group in a single or multiple populations (when `nPop` > 1)

propE.control	scalar or vector of proportion of control trios in the environmental exposure group in a single or multiple populations (when nPop > 1)
Einteraction	string indicating what variable environmental effects interact with (one of "Im", "If", "C", or "M")
nPop	number of populations to simulate for population stratification, see details for information on specifying other parameters when there are multiple strata
maf	scalar or vector of the minor allele frequency proportion in a single or multiple populations (when nPop > 1)
prev.byPop	prevalence of cases in each sub population (vector with length equal to nPop, ignored if nPop = 1)
prop.byPop	proportion of each sub population, must sum to 1 (vector with length equal to nPop, ignored if nPop = 1)
Fst	F parameter for estimation of maf when sub-population level maf's are unknown (ignored when nPop = 1 or when maf has the same length as nPop), see details for more information

Details

To simulate the counts, first the total number of case trios (nCases) and control trios (nControl) are partitioned into the nPop sub-populations by randomly sampling from respective multinomial distributions, with probabilities calculated using prop.byPop and prev.byPop as follows:

$$\Pr(X = i, E = 1 \mid D = 1) = \frac{\alpha_i \varepsilon_{i,1} \omega_i}{\alpha_1 \omega_1 + \alpha_2 \omega_2},$$

$$\Pr(X = i, E = 0 \mid D = 1) = \frac{\alpha_i (1 - \varepsilon_{i,1}) \omega_i}{\alpha_1 \omega_1 + \alpha_2 \omega_2},$$

$$\Pr(X = i, E = 1 \mid D = 0) = \frac{(1 - \alpha_i) \varepsilon_{i,0} \omega_i}{(1 - \alpha_1) \omega_1 + (1 - \alpha_2) \omega_2},$$

$$\Pr(X = i, E = 0 \mid D = 0) = \frac{(1 - \alpha_i)(1 - \varepsilon_{i,0}) \omega_i}{(1 - \alpha_1) \omega_1 + (1 - \alpha_2) \omega_2},$$

where $X = i$ is the i-th sub-population, α_i is the i-th element of prev.byPop, $\varepsilon_{i,j}$ i-th element of propE (when $j = 1$) or propE.control (when $j = 0$), and ω_i is the i-th element of prop.byPop.

If only a single aggregated maf value is provided when there are multiple sub-populations, maf may be estimated using the model from Balding and Nichols (1995). This requires an F_{ST} parameter to be provided, with estimates of maf being obtained by sampling from a beta distribution with shape parameters

$$\alpha = \text{MAF} \times \left(\frac{1}{F_{ST}} - 1 \right),$$

$$\beta = (1 - \text{MAF}) \times \left(\frac{1}{F_{ST}} - 1 \right),$$

where MAF is the provided maf.

If nPop = 1, this step is skipped as everyone will be in the same sub-population. Note that this method of simulating population stratification assumes D and E are conditionally independent (i.e., V = c(1,1,1)).

Distributions of counts in each partition based on the different genotype, exposure, and disease status categories are then independently sampled based on the risk of disease (from the given `R`, `S`, `V`, `Im`, and `If` values) and mating type probabilities (from `mtCoef`).

`R`, `S`, and `V` are vectors of 3 elements, where the first element should be 1, acting as a baseline for when M, C, or $M \times E$ are 0 respectively. The following two numbers represent the multiplicative increase to risk when the respective genotype is 1 or 2. `Im` and `If` are the imprinting parameters, representing the multiplicative increase to risk when the minor allele is maternally or paternally inherited.

Mating type probabilities are simulated using the mating asymmetry parameterization of Bourgey et al. (2011). Here, `mtCoef` is a vector of 3 elements representing (C_1, C_2, C_4). We begin by assigning each $\Pr(M, F)$ assuming the population is in Hardy-Weinberg equilibrium, with probabilities obtained from the given `maf`. Letting

$$\mu_1 = \Pr(M = 2, F = 1) = \Pr(M = 1, F = 2),$$
$$\mu_2 = \Pr(M = 2, F = 0) = \Pr(M = 0, F = 2),$$
$$\mu_4 = \Pr(M = 1, F = 0) = \Pr(M = 0, F = 1),$$

we reassign these particular mating pairs by setting

$$\Pr(M = 2, F = 1) = \mu_1 C_1,$$
$$\Pr(M = 1, F = 2) = \mu_1 (2 - C_1),$$
$$\Pr(M = 2, F = 0) = \mu_2 C_2,$$
$$\Pr(M = 0, F = 2) = \mu_2 (2 - C_2),$$
$$\Pr(M = 1, F = 0) = \mu_4 C_4,$$
$$\Pr(M = 0, F = 1) = \mu_4 (2 - C_4).$$

To view the data frame with underlying simulation probabilities, counts for either parent of origin case when mother, father, and child are all, heterozygous, and by sub-population when simulating population stratification, use fullview.

Value

An object of class `TriLLIEM.sim`, a data frame with columns for:

`"type"` Index for the category corresponding to maternal (`"M"`), paternal (`"F"`), and child (`"C"`) genotypes.

`"mt_MS"` Mating type category for maternal and paternal genotypes under a mating symmetry model.

`"mt_MaS"` Mating type category for maternal and paternal genotypes under a model that does not assume mating symmetry.

`"M"` Maternal genotype.

`"F"` Paternal genotype.

`"C"` Child genotype.

`"E"` Binary variable for if environmental effects are present (1) or not present (0).

`"D"` Case child (1)/Control child (0).

`"count"` Number of triads falling under the specified category based on `"M"`, `"F"`, `"C"`, `"E"`, `"D"`.

References

Balding DJ, Nichols RA (1995). "A method for quantifying differentiation between populations at multi-allelic loci and its implications for investigating identity and paternity." *Genetica*, **96**(1–2), 3–12. ISSN 1573-6857, doi:10.1007/bf01441146, http://dx.doi.org/10.1007/BF01441146.

Bourgey M, Healy J, Saint-Onge P, Massé H, Sinnett D, Roy-Gagnon M (2011). "Genome-wide detection and characterization of mating asymmetry in human populations." *Genetic Epidemiology*, **35**(6), 526-535. doi:10.1002/gepi.20602, https://onlinelibrary.wiley.com/doi/pdf/10.1002/gepi.20602, https://onlinelibrary.wiley.com/doi/abs/10.1002/gepi.20602.

See Also

example_dat4R, fullview

Examples

```
## Maternal effect of 2, and paternal imprinting of 3.
simulateData(S = c(1, 2, 4), If = 3)

## Paternal imprinting by environment interaction of 1.5.
simulateData(V = c(1, 1.5, 1.5^2), propE = 0.3, Einteraction = "If")

## Maternal gene environment interaction of 1.6 with controls
simulateData(nControl = 1000, V = c(1, 1.6, 1.6^2), propE = 0.3, Einteraction = "M")

## Null model with 3 different sub-populations
simulateData(
    nPop = 3,
    maf = c(0.1, 0.2, 0.3),
    prev.byPop = c(0.2, 0.1, 0.4),
    prop.byPop = c(0.3, 0.3, 0.4)
)

## Null model with 2 different sub-populations, environmental exposures, and controls
simulateData(
    nControl = 1000,
    propE = c(0.1, 0.4),
    propE.control = c(0.2, 0.2),
    nPop = 2,
    maf = c(0.3, 0.4),
    prev.byPop = c(0.2, 0.3),
    prop.byPop = c(0.6, 0.4)
 )
```

---

summary.TriLLIEM *Summary function for TriLLIEM functions*

Description

Summary function for TriLLIEM functions

Usage

```
## S3 method for class 'TriLLIEM'
summary(object, ...)
```

Arguments

`object`	an object of class "`TriLLIEM`", resulting from a call to TriLLIEM.
`...`	arguments passed to or from other methods.

Details

Due to TriLLIEM using the EM algorithm when fitting imprinting effects, calculation of residuals is modified compared to in summary.glm, where the original data (`object$y_initial`) is used instead of `object$y` to ensure the correct residuals are computed. See TriLLIEM for more details.

Value

A list with the same components as those returned by summary.glm, but with the addition of the following:

terms the same component from `object`, always included for "`TriLLIEM`" objects.

aic the same component from `object`, always included for "`TriLLIEM`" objects

EM_iter the same component from `object`, always included for "`TriLLIEM`" objects

Examples

```
res <- TriLLIEM(mtmodel = "HWE", effects = c("C", "M", "Im"), dat = example_dat4R)
summary(res)
```

`TriLLIEM` *Fit the log-linear model to trio data*

Description

This function is used to fit the user-specified log-linear model to trio count data.

Usage

```
TriLLIEM(
  mtmodel = "MS",
  effects = c("C", "M"),
  dat,
  includeE = FALSE,
  Estrat = FALSE,
  Eanova = FALSE,
  includeD = FALSE,
  Minit = 0.5,
  max.iter = 30,
  EM.diag = FALSE
)
```

Arguments

mtmodel	Mating type model to use in the analysis, can be "HWE" for Hardy-Weinberg Equilibrium, "MS" for Mating Symmetry (default), and "MaS" for Mating Asymmetry.
effects	A vector listing the effects, as strings, to include in the model. Effects can include: "C" Child effects. "M" Maternal effects. "Im" Maternal imprinting effects. "If" Paternal imprinting effects. "E:C" Child gene environment interactions. "E:M" Maternal gene environment interactions. "E:Im" Maternal imprinting by environment interactions. "E:If" Paternal imprinting by environment interactions. Default is c("C", "M").
dat	A data frame with triad data, with the formatting of example_dat4R.
includeE	A logical value indicating whether to include environment interaction effects. If set to "FALSE", any exposed counts in dat are combined with the respective unexposed count (treating dat as if all E = 0). Default is FALSE.
Estrat	A logical value indicating whether to use a stratified approach for environmental interactions. Default is FALSE. See details for more information.
Eanova	A logical value indicating if this is for the sake of running anova to compare a model with environmental interactions to a model that does not include environmental interactions. Should be left as "FALSE" if not for this purpose, as the degrees of freedom will be incorrect. See details for more information. Default is FALSE.
includeD	A logical value indicating whether to use the hybrid model with controls. If set to "FALSE", any control trios will be removed from the data set prior to analysis. Default is FALSE.

Minit	Initial value for the proportion of triple heterozygote (M=1, F=1, C=1) category where the '1' allele is passed from the mother to child. This is used to initialize the EM algorithm. Default is 0.5.
max.iter	Maximum number of iterations for the EM algorithm. Default is 30.
EM.diag	A logical value indicating whether to show diagnostic messages for the EM algorithm. Default is FALSE.

Details

Fits the specified log-linear model of Weinberg et al. (1998) to dat using R's glm framework. This includes Weinberg and Umbach (2005)'s hybrid model allowing for control triads. When imprinting effects ("Im" or "If") are included, an EM algorithm is run to estimate the counts of $(M, F, C) = (1, 1, 1)$ triads which are maternally and paternally inherited. Normally, if this was done using the glm function, the computed likelihoods would use these estimated counts, which is incorrect. This function, alongside its methods like summary.TriLLIEM and anova.TriLLIEM, performs the EM algorithm while using the original observed counts to correctly obtain likelihoods.

The model formula supplied to glm is made up of the genetic effects supplied in effects, alongside several nuisance parameters. These include:

- Mating type parameters, dependent on the specified mtmodel,
- D, if includeD = TRUE,
- E:D, if includeD = TRUE and includeE = TRUE,
- Mating type parameter by E interactions, if includeE = TRUE; E, if includeE = TRUE and mtmodel = "HWE" (both necessary as shown in Shin et al. (2010)). These nuisance parameters are omitted when printing the function output, but may be viewed by using the coef function on the output.

Estrat = TRUE forces every every listed effect in effects to have its gene environment interaction included in the model, regardless of whether the user has specified them explicitly or not. This essentially stratifies the model by E, and is useful for replicating the stratified models necessary for analyzing gene environment interactions in Haplin (Gjessing and Lie 2006) and EMIM (Howey and Cordell 2012).

Eanova = TRUE allows the model to be fit when $E \neq 0$ rows are present but includeE = FALSE, without modifying these exposed rows. This is only useful for using anova.TriLLIEM to determine if models with gene-environment interactions are statistically different from models without gene-environment interactions. Since this option keeps these $E \neq 0$ categories without using the E parameter though, the fitted model will use incorrect degrees of freedom in its significance tests, and hence should not be used for any inferences besides this very specific case of model comparison.

All TriLLIEM objects are of family poisson_em, a modified version of the poisson family to account for additional data created by the EM algorithm when computing residuals and AIC.

Value

An object of class "TriLLIEM", which inherits from class "glm" and has the same components as the output of glm, with the following modifications:

df.residual if imprinting effects are specified, subtracted by the number of additional rows introduced by the EM algorithm.

`df.null` if imprinting effects are specified, subtracted by the number of additional rows introduced by the EM algorithm.

`EM_iter` number of EM algorithm iterations before convergence.

`y_initial` initial supplied data frame, `dat`, before addition of rows by EM.

`grp` list of vectors, with each vector containing indices of grouped rows in the data after the EM algorithm is applied (`y`). Aggregating these grouped rows by sum will yield `y_initial`.

References

Gjessing HK, Lie RT (2006). "Case-parent triads: estimating single- and double-dose effects of fetal and maternal disease gene haplotypes." *Annals of Human Genetics*, **70**(Pt 3), 382–396.

Howey R, Cordell HJ (2012). "PREMIM and EMIM: tools for estimation of maternal, imprinting and interaction effects using multinomial modelling." *BMC Bioinformatics*, **13**(1). ISSN 1471-2105, doi:10.1186/1471210513149, http://dx.doi.org/10.1186/1471-2105-13-149.

Shin J, McNeney B, Graham J (2010). "On the Use of Allelic Transmission Rates for Assessing Gene-by-Environment Interaction in Case-Parent Trios." *Annals of Human Genetics*, **74**(5), 439-451. doi:10.1111/j.14691809.2010.00599.x, https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1469-1809.2010.00599.x, https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1469-1809.2010.00599.x.

Weinberg CR, Umbach DM (2005). "A Hybrid Design for Studying Genetic Influences on Risk of Diseases with Onset Early in Life." *The American Journal of Human Genetics*, **77**(4), 627–636. ISSN 0002-9297, doi:10.1086/496900, http://dx.doi.org/10.1086/496900.

Weinberg CR, Wilcox AJ, Lie RT (1998). "A log-linear approach to case-parent-triad data: assessing effects of disease genes that act either directly or through maternal effects and that may be subject to parental imprinting." *American Journal of Human Genetics*, **62**(4), 969–978.

Examples

```
res1 <- TriLLIEM(mtmodel = "HWE", effects = c("C", "M", "Im"), dat = example_dat4R)
summary(res1)

dat <- simulateData(
    nControl = 1000,
    propE = c(0.1, 0.4),
    propE.control = c(0.2, 0.2),
    nPop = 2,
    maf = c(0.3, 0.4),
    prev.byPop = c(0.2, 0.3),
    prop.byPop = c(0.6, 0.4)
)
## Obtain the non-stratified and stratified models and compare them via anova
res2 <- TriLLIEM(
    mtmodel = "HWE",
    effects = c("C", "M", "Im", "E:Im"),
    dat = dat,
    includeE = TRUE,
```

```
    includeD = TRUE
)
res3 <- TriLLIEM(
    mtmodel = "HWE",
    effects = c("C", "M", "Im", "E:Im"),
    dat = dat,
    includeE = TRUE,
    Estrat = TRUE,
    includeD = TRUE
)
anova(res2, res3)

## Compare non-stratified model to a model without E by setting Eanova = TRUE
res4 <- TriLLIEM(
    mtmodel = "HWE",
    effects = c("C", "M", "Im"),
    dat = dat,
    Eanova = TRUE,
    includeD = TRUE
)

anova(res2, res4)
```

Index