Package 'ecoreg'

August 24, 2017

Title Ecological Regression using Aggregate and Individual Data

Version 0.2.2

Date 2017-08-24

Author Christopher Jackson <chris.jackson@mrc-bsu.cam.ac.uk>

Description Estimating individual-level covariate-outcome associations using aggregate data (``ecological inference'') or a combination of aggregate and individual-level data (``hierarchical related regression'').

Maintainer Christopher Jackson <chris.jackson@mrc-bsu.cam.ac.uk>

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2017-08-24 15:04:43 UTC

R topics documented:

`eco` *Ecological regression using aggregate and/or individual data*

Description

Estimation of an underlying individual-level logistic regression model, using aggregate data alone, individual-level data alone or a combination of aggregate and individual-level data. Any number number of covariates can be included in the individual-level regression. Covariates can be binary or categorical, expressed as proportions over the group, or normally-distributed, expressed as within-area means and optional covariances. A general formula for group-level (contextual) effects can also be supplied.

Usage

```
eco(formula, binary, categorical, normal, iformula, data, idata, groups, igroups,
strata, istrata, pstrata, cross=NULL, norm.var=NULL, random=FALSE,
pars, fixed=FALSE, model = c("marginal","conditional"),
outcome=c("binomial","poisson"), gh.points=10, iter.adapt=5, ...)
```

Arguments

formula	A model formula containing the group-level binomial response on the left-hand side, and general group-level covariates on the right-hand side. For example, `cbind(n.cases, population) ~ mean.income + deprivation.index` If `formula` is not specified, then there is assumed to be only individual-level data, and `iformula` should be supplied.
binary	An optional model formula with an empty left-hand side. The right-hand side should contain the names of any group-level proportions, which are to modelled as individual-level binary predictors of the response given in `formula`. For example, `~ p.smokers + p.nonwhite + p.unemployed`
categorical	An optional list of matrices or data frames. Each element corresponds to a categorical covariate. Each element has the same number of rows as the aggregate data, and number of columns corresponding to the number of levels of the categorical covariate. The cells give the number or proportion of individuals in the area in each category. These will be modelled as individual-level predictors of the response given in `formula`.
normal	An optional model formula with an empty left-hand side. The right-hand side should list variables containing the group-level means of normally-distributed covariates. These will be modelled as individual-level predictors of the response given in `formula`. For example `~ pollution + income`.
iformula	A model for the corresponding individual-level data. The individual-level binary response should be on the right-hand side, and the individual-level covariates should be on the left-hand side. They should represent the same covariates, in the same order, as given in `formula` and `binary` respectively. However they need not have the same names. For example `outcome ~ mean.income + deprivation.index + smoking + nonwhite + unemployed`. If `iformula` is not specified, then there is assumed to be only aggregate data, and `formula` should be supplied.
data	Data frame containing the group-level variables given in `formula` and `binary`.
idata	Data frame containing the individual-level variables given in `iformula`.
groups	A group-level variable containing the group identifiers to be matched with the groups given in `igroups`. Defaults to the row numbers of the aggregate data. Only necessary if the model includes random group effects.
igroups	An individual-level variable containing the group identifiers of the individual-level data to be matched with the groups given in `groups`. Only necessary if the model includes random group effects.

`strata` A matrix with the same number of rows as the aggregate data. Rows representing groups, and columns representing strata occupancy probabilities, often estimated as observed occupancy proportions. The relative risks for the strata will be included as fixed offsets in the underlying logistic regression, using the probabilites supplied in `pstrata`. This is to save the computational burden of estimating the "nuisance" strata-specific risks from the data.

`istrata` A variable containing the individual-level variable indicating the stratum an individual occupies. This should be a factor with the levels corresponding to the columns of the matrix `strata`.

`pstrata` A vector with one element for each stratum, giving the assumed baseline outcome probabilities for the strata.

`cross` A matrix giving the joint within-area distribution of all the covariates supplied in `binary` and `categorical` and any strata. This should have the same number of rows as the aggregate data, and number of columns equal to the product of the numbers of levels of the covariates and strata, for example 2^n if there are n binary covariates. Each cell gives the proportion of individuals in the area occupying a category defined by a unique combination of the covariates. The categories are given in the order

column 1: covariate 1 absent, covariate 2 absent, ..., covariate n-1 absent, covariate n absent
column 2: covariate 1 present, covariate 2 absent, ..., covariate n-1 absent, covariate n absent
column 3: covariate 1 absent, covariate 2 present, ..., covariate n-1 absent, covariate n absent
column 4: covariate 1 present, covariate 2 present, ..., covariate n-1 absent, covariate n absent
etc.

(assuming n binary covariates, with the obvious generalisation for categorical covariates) If `strata` are used, these are taken as covariate n+1.

`norm.var` A data frame, matrix or list, supplying the within-area covariances of the continuous covariates.

If `norm.var` is a data frame or matrix, then the continuous covariates are assumed to be independent within areas. It should have rows corresponding to areas, columns corresponding to continuous covariates, each cell giving the within-area standard deviation of the covariate.

If `norm.var` is a list, then it should have the same number of elements as the number of areas, and each element should be the within-area covariance matrix of the continuous covariates.

`norm.var` can also be the name of a variable in `data` which contains the standard deviation of a single continuous covariate.

`random` If `TRUE` then a normally-distributed random group-level intercept, with zero mean, is also included in the model.

`pars` Vector of initial values of the model parameters, given in the following order:
logit-scale intercept,
coefficients for group-level covariates,
coefficients for individual-level covariates,

random effects standard deviation.

If not supplied, the initial values are 0 for all covariate effects, 1 for the random effects standard deviation. The intercept is initialised to the logit mean outcome proportion over groups from the aggregate data.

`fixed` If `TRUE` then `eco` just calculates the likelihood with all parameters are fixed at their initial values.

`model` If "marginal" then the ecological group-level risk is based on integrating over binary individual-level covariates. This is suitable if the aggregate exposures are estimated using a survey of individuals in the area. If "conditional" then the binary individual-level covariates are conditioned on, and the group-level risk is the normal approximation model described by Wakefield (2004). This is suitable if the aggregate exposures are estimated using a full population census.

`outcome` Distribution of the aggregate outcome, by default "binomial". `outcome="poisson"` can be specified for rare outcomes.

`gh.points` Number of points for Gauss-Hermite numerical integration in the random effects model.

`iter.adapt` Number of adaptive iterations to estimate the mode and scale for Gauss-Hermite numerical integration in the random-effects model.

`...` Arguments passed to `optim`.

Details

Individual data are simply modelled by a logistic regression.

Aggregate outcomes are modelled as binomial, with area-level risk obtained by integrating the underlying individual-level logistic regression model over the within-area distribution of the covariates.

The model for combined individual and aggregate data shares the same coefficients between the individual and aggregate components.

Aggregate data alone can be sufficient for inference of individual-level relationships, provided the between-area variability of the exposures is large compared to the within-area variability.

When there are several binary covariates, it is usually advisable to account for their within-area distribution, using `cross`.

See Jackson et al. (2006,2008) for further details.

Value

A list with components:

`call` The call to `eco`.

`lik` Minus twice the log-likelihood at the estimates.

`ors.ctx` Matrix of estimated odds ratios and 95% confidence intervals for the area-level covariates.

`ors.indiv` Matrix of estimated odds ratios and 95% confidence intervals for the individual-level covariates.

random	The estimated random-effects standard deviation.
mod	A list of constants describing the model and data (not useful to end users).
corrmat	The correlation matrix of the maximum likelihood estimates (on the optimized scale, for example log odds ratios for covariates).

Author(s)

C. H. Jackson <chris.jackson@mrc-bsu.cam.ac.uk>

References

C. H. Jackson, N. G. Best, and S. Richardson. (2006) *Improving ecological inference using individual-level data.* Statistics in Medicine, 25(12): 2136-2159.

C. H. Jackson, N. G. Best, and S. Richardson. (2008) *Hierarchical related regression for combining aggregate and survey data in studies of socio-economic disease risk factors.* Journal of the Royal Statistical Society, Series A, 171(1):159-178.

J. Wakefield. (2004) *Ecological inference for 2 x 2 tables* (with discussion). Journal of the Royal Statistical Society, Series A, 167(3) 385–445.

J. Wakefield and R. Salway. (2001) *A statistical framework for ecological and aggregate studies*. Journal of The Royal Statistical Society, Series A, 164(1):119–137, 2001.

See Also

sim.eco

Examples

```
## Simulate some aggregate data and some combined aggregate and
## individual data.
ng <- 50
N <- rep(100, ng)
set.seed(1)
ctx <- cbind(deprivation = rnorm(ng), mean.income = rnorm(ng))
phi <- cbind(nonwhite = runif(ng), smoke = runif(ng))
sim.df <- as.data.frame(cbind(ctx, phi))
mu <- qlogis(0.05)  ## Disease with approximate 5% prevalence

## Odds ratios for group-level deprivation and mean imcome
alpha.c <- log(c(1.01, 1.02))
## Odds ratios for individual-level ethnicity and smoking
alpha <- log(c(1.5, 2))

sim1 <- sim.eco(N, ctx=~deprivation+mean.income, binary=~nonwhite+smoke,
                data = sim.df,  mu=mu, alpha.c=alpha.c, alpha=alpha)
sim2 <- sim.eco(N, ctx=~deprivation+mean.income, binary=~nonwhite+smoke,
                data = sim.df,  mu=mu, alpha.c=alpha.c, alpha=alpha, isam=7)

## Fit the model to recover the simulated odds ratios.
```

```
aggdata <- as.data.frame(cbind(y=sim1$y, sim.df))
agg.eco <- eco(cbind(y, N) ~ deprivation + mean.income,
               binary = ~ nonwhite + smoke,  data = aggdata)
agg.eco

## Combining with individual-level data
## doesn't improve the precision of the estimates.

agg.indiv.eco <- eco(cbind(y, N) ~ deprivation + mean.income,
               binary = ~ nonwhite + smoke,
               iformula = y ~ deprivation + mean.income + nonwhite + smoke,
               data = aggdata, idata=sim2$idata)
agg.indiv.eco

## However, suppose we have much lower between-area variance in the
## mean covariate value.

phi <- cbind(nonwhite = runif(ng, 0, 0.3), smoke = runif(ng, 0.1, 0.4))
sim.df <- as.data.frame(cbind(ctx, phi))
sim1 <- sim.eco(N, ctx=~deprivation+mean.income, binary=~nonwhite+smoke,
                data = sim.df,  mu=mu, alpha.c=alpha.c, alpha=alpha)
sim2 <- sim.eco(N, ctx=~deprivation+mean.income, binary=~nonwhite+smoke,
                data = sim.df,  mu=mu, alpha.c=alpha.c, alpha=alpha, isam=10)
aggdata <- as.data.frame(cbind(y=sim1$y, sim.df))

## The aggregate data now contain little information about the
## individual-level effects, and we get biased estimates of the true
## individual model.
agg.eco <- eco(cbind(y, N) ~ deprivation + mean.income,
               binary = ~ nonwhite + smoke,  data = aggdata)
agg.eco

## We need individual-level data to be able to estimate the
## individual-level effects accurately.
agg.indiv.eco <- eco(cbind(y, N) ~ deprivation + mean.income,
               binary = ~ nonwhite + smoke,
               iformula = y ~ deprivation + mean.income + nonwhite + smoke,
               data = aggdata, idata=sim2$idata)
agg.indiv.eco

## But then why not just study the individual data?  Combining with
## aggregate data improves precision.
indiv.eco <- eco(iformula = y ~ deprivation + mean.income + nonwhite + smoke,
               idata=sim2$idata)
indiv.eco
```

`gauss.hermite` *Calculate Gauss-Hermite Quadrature Points*

Description

`gauss.hermite` calculates the Gauss-Hermite quadrature values for a specified number of points. From the **rmutil** package by Jim Lindsey (`http://luc.ac.be/~jlindsey/rcode.html`)

Usage

```
gauss.hermite(points, iterlim=50)
```

Arguments

`points`	The number of points.
`iterlim`	Maximum number of iterations in Newton-Raphson.

Value

`gauss.hermite` returns a two-column matrix containing the points and their corresponding weights.

Author(s)

J.K. Lindsey

Examples

```
gauss.hermite(10)
```

`integrate.gh` *Univariate Gauss-Hermite integration*

Description

Computes the integral of a univariate function, or several univariate functions simultaneously, using Gauss-Hermite quadrature.

Usage

```
integrate.gh(h, n=1, points = 10, mu = 0, scale = 1, ...)
```

Arguments

`h`	The function to be integrated. May either have a scalar first argument and return a scalar result, or have a first argument of length `n` and return a vector of `n` results, corresponding to `n` independent functions.
`n`	The dimension of the result returned by `h`.
`points`	Number of Gauss-Hermite quadrature points.
`mu`	Mode of the function, to centre the quadrature points around.
`scale`	Scale of the quadrature points.
`...`	Other arguments to be passed to `h`.

Details

The integral is more accurate if the standard quadrature points are shifted and scaled to match the mode and scale of $g(x)$, that is the objective function divided by the standard normal density. The scale is estimated by $1/\sqrt{-H}$, where H is the Hessian at the maximum of $g(x)$.

Value

The integral of $h(x)$ between `-Inf` and `Inf`, of length `n`. In the usual application of Gauss-Hermite quadrature, $h(x)$ is equivalent to a function $g(x)\phi(x)$, where $\phi(x)$ is the standard normal density function.

Author(s)

C. H. Jackson <chris.jackson@mrc-bsu.cam.ac.uk>

The Gauss-Hermite polynomial values and weights are calculated using the `gauss.hermite` function copied from the **rmutil** package by J. K. Lindsey.

References

Liu, Q. and Pierce, D. A. (1994) *A note on Gauss-Hermite quadrature*. Biometrika, 81 (624-629)

See Also

`gauss.hermite`

Examples

```
## Want the integral of h over the real line
g <- function(x) 4 * exp( - ((1 - x)^2 + 1))
h <- function(x) g(x) * dnorm(x)
integrate(h, -Inf, Inf)
integrate.gh(h)
## Not very accurate with default 10 points. Either use more quadrature points,
integrate.gh(h, points=30)
## or shift and scale the points.
opt <- nlm(function(x) -g(x), 0, hessian=TRUE)
integrate.gh(h, mu=opt$estimate, scale=1/sqrt(opt$hessian))
```

`sim.eco` *Simulate ecological data and samples of individual-level data*

Description

Simulate ecological data and samples of individual-level data from an individual-level logistic regression model, depending on given binary, categorical or normally-distributed covariates.

Usage

```
sim.eco(N, ctx, binary, m, data=NULL, S=0, cross=NULL, covnames, ncats,
mu, alpha.c=0, alpha=0, beta=0, sig=0, strata, pstrata, isam = 0)
```

Arguments

N	Vector of population sizes, one for each group.
ctx	A model formula containing names of group-level, or contextual, covariates on the right-hand side.
binary	A model formula containing names of individual-level binary covariates on the right-hand side.
data	Data frame containing the group-level variables given in ctx. It should also contain the variables given in binary, interpreted as proportions of individuals exposed to each of the binary covariates.
m	A data frame with length(N) rows, containing the within-area means of a set of normally-distributed continuous covariates.
S	A data frame with length(N) rows, containing the within-area standard deviations of a set of normally-distributed continuous covariates. For the moment they are assumed to be independent.
cross	A matrix of cross-classifications of individuals in the area between categories of multiple binary or categorical covariates, defined in the same way as in eco. If this is not supplied, the binary covariates are assumed to be independent, and the probability of an individual having a certain combination of covariates is calculated as the product of the relevant marginal probabilities.
covnames	Vector of names of the covariates, if cross is supplied. Otherwise the names are taken from binary.
ncats	Numeric vector of the number of levels of the covariates used in cross.
mu	Regression intercept on the logit scale.
alpha.c	Vector of coefficients for the group-level covariates in the underlying logistic regression, corresponding to the columns of ctx.
alpha	Vector of coefficients for the individual-level binary covariates, corresponding to the columns of phi. Interactions are not currently supported.
beta	Vector of coefficients for the individual-level continuous covariates, corresponding to the columns of m or S. Interactions are not currently supported.
sig	Random-effects standard deviation.
strata	A matrix with rows representing groups, and columns representing strata occupancy probabilities.
pstrata	A vector with one element for each stratum, giving the assumed baseline outcome probabilities for the strata. The logits of pstrata are used as offsets in the logistic regression.
isam	Number of individuals per group to retain in the individual-level data.

Value

A list with components:

y	The simulated aggregate-level response, one for each group.
idata	A data frame containing the retained individual-level samples. The grouping indicator (with values 1 to length(N)) is named group, the response variable is named y, the group-level covariates are copied from ctx, and the binary covariates, with values 1 or 0, are individual-level equivalents of the proportions given in phi.

Author(s)

C. H. Jackson <chris.jackson@mrc-bsu.cam.ac.uk>

See Also

eco

Examples

```
N <- rep(50, 20)
ctx <- cbind(deprivation = rnorm(20), mean.income = rnorm(20))
phi <- cbind(nonwhite = runif(20), smoke = runif(20))
sim.df <- as.data.frame(cbind(ctx, phi))
mu <- qlogis(0.05)  ## Disease with approximate 5% prevalence
## Odds ratios for group-level deprivation and mean imcome
alpha.c <- c(1.01, 1.02)
## Odds ratios for individual-level ethnicity and smoking
alpha <- c(1.5, 2)
sim.eco(N, ctx = ~ deprivation + mean.income, binary = ~ nonwhite +
        smoke, data=sim.df, mu=mu, alpha.c=alpha.c, alpha=alpha)
sim.eco(N, ctx = ~ deprivation + mean.income, binary = ~ nonwhite +
        smoke, data=sim.df, mu=mu, alpha.c=alpha.c, alpha=alpha, isam=3)
```

---

tapplysum.fast *Simplified fast group sums*

Description

A hack to speed up tapply(x, group, sum) for the special case where x is sorted by group.

Usage

```
tapplysum.fast(x, groups)
```

Arguments

x	A numeric vector.
groups	A grouping factor.

Details

Works by computing the cumulative sum of x and taking the difference at the indices where the groups change. Standard tapply can be slow when there are a large number of groups, due to the overhead of factor manipulation.

Value

Vector containing the group sums of x.

Author(s)

C. H. Jackson <chris.jackson@mrc-bsu.cam.ac.uk>

See Also

tapply

Examples

```
x <- factor(rep(1:1000, each=100))
y <- rnorm(1000*100)
system.time(tapply(y, x, sum))
system.time(tapplysum.fast(y, x))
```

Index