Package ‘partR2’

October 14, 2022

Title Partitioning R2 in GLMMs

Version 0.9.1

Depends R (>= 3.5.0)


License GPL (>= 2)

URL https://github.com/mastoffel/partR2

BugReports https://github.com/mastoffel/partR2/issues

Imports methods, stats, lme4 (>= 1.1-21), pbapply (>= 1.4-2), dplyr (>= 0.8.3), purrr (>= 0.3.3), rlang (>= 0.4.2), tibble (>= 2.1.3), magrittr (>= 1.5), ggplot2 (>= 3.3.0), tidyr (>= 1.1)

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Suggests testthat, future, furrr, knitr, rmarkdown, patchwork, covr

VignetteBuilder knitr

NeedsCompilation no

Author Martin A. Stoffel [aut, cre], Shinichi Nakagawa [aut], Holger Schielzeth [aut]

Maintainer Martin A. Stoffel <martin.adam.stoffel@gmail.com>

Repository CRAN

Date/Publication 2021-01-18 16:30:04 UTC
The partR2 package provides a simple way to estimate R2 in mixed models fitted with lme4 as well as part (semi-partial) R2 for specific predictors and combinations of predictors, among other several other statistics. Here is an overview:

Details

- Marginal and conditional R2 for LMMs and GLMMs.
- Part (semi-partial) R2 which estimate the explained variance for specific predictors and combinations of predictors.
- Structure coefficients (SC). SC are the correlation between a predictor and the predicted response (called the linear predictor), independent of the other predictors.
- Inclusive R2 (IR2), which estimate the the total variance explained by a predictor independent of other predictors. IR2 is estimated with SC^2 * R2_full_model.
- Beta weights, which are standardised regression coefficients. If beta is a model estimate for variable x, and y is the response, then the beta weight is beta * (sd(x)/sd(y).

The package has one main function `partR2` which takes a fitted model from lme4. At the moment, Gaussian, Poisson and binomial models are supported. For Poisson and non-binary binomial models, partR2 adds an observational level random effect to model additive overdispersion (if an olre is not fitted already).

The `summary.partR2` function provides an extended summary with R2s, semi-partial R2s, model estimates and structure coefficients. The `forestplot` function provides a means of plotting the results.
biomass dataset

Description

biomass dataset

Details

This is an simulated dataset about grassland biodiversity and biomass.

In brief the imaginary sampling design of the simulated dataset is as follows. Invertebrates were sampled once every year over 10 successive years ('Year') from 20 different populations ('Population'). For each sample, the temperature ('Temperature') and ('Precipitation') were measured and overall species diversity ('SpeciesDiversity') and biomass were recorded ('Biomass').

forestplot

Plot a partR2 object

Description

Forestplot of the partR2 results

Usage

```r
forestplot(
  x,
  type = c("R2", "BW", "SC", "IR2", "Ests"),
  line_size = 0.5,
  text_size = 12,
  point_size = 3
)
```
Arguments

- x: A partR2 object.
- type: Plot either "R2" or "SC" or "Ests" for R2s, structure coefficients or model estimates.
- line_size: Controls size of all lines in the forestplot. Defaults to 0.5 which usually looks good.
- text_size: Base text size, default is 12.
- point_size: Point size, default is 3.

Grasshoppers dataset

Description

Grasshoppers dataset

Details

This is a real dataset from grasshoppers.

This dataset contains data on spatial variation in color morph ratios in a color-polymorphic species of grasshopper (Dieker et al 2018). Individuals of this species occur either in green or a brown color variant and the dataset contains counts of brown and green individuals (separated for females and males) from 42 sites sampled in the field. All `Bio` variables describe various aspects of ecologically relevant climatic conditions (see Karger et al. 2017).

References


GuineaPigs

GuineaPigs dataset

Description

GuineaPigs dataset

Details

This is a real dataset from guinea pigs.

The dataset contains testosterone measurements (‘Testo’) of 31 male guinea pigs, each measured at 5 time points. (age between 120 and 240 days at 30-day intervals). As covariates the dataset contains the time point of measurement (‘Time’) and a rank index derived from behavioral observations (‘Rank’) around the time of measurement (see Mutwill et al. in prep. for details).

mergeR2

Merge partR2 objects to combine R2s for main effects and interactions

Description

The function merges partR2 object based on a full model with interactions with a partR2 object based on a reduced model without interaction. The reduced model is used to infer main effect semi-partial R2s. This function essentially takes over the complete partR2 object for the full model and adds semi-partial R2s which have been calculated based on the reduced model and are not already present in the full model partR2 object (which can be main effects). The function also combines the bootstrap estimates, accessible with partR2_obj$R2_boot.

Usage

mergeR2(R2_full, ...)

Arguments

R2_full partR2 object for the full model, with the interaction (but not the main effects) in partvars.

... other partR2 objects, which do not contain the interaction so that the semi-partial R2s for the main effects could be calculated.

Details

This function is experimental and should be used with caution. See vignette or paper on how to use it to obtain semi-partial R2s for main effects which are also involved in interactions.
Value

Returns an object of class partR2, which takes most components from the full model except for semi-partial R2s.

Examples

data(biomass)
library(lme4)

# scale data
biomass[] <- lapply(biomass, function(x) if (is.double(x)) scale(x) else x)

# Full model
mod_full <- lmer(Biomass ~ Year + Temperature * Precipitation + SpeciesDiversity + (1|Population),
                 data = biomass)
# Semi-partial R2 for interaction and all other predictors of interest
(R2_full <- partR2(mod_full, partvars = c("Temperature:Precipitation", "SpeciesDiversity", "Year"),
                   data = biomass))

# model without interaction to get main effect semi-partial R2s
mod_noIA <- lmer(Biomass ~ Year + Temperature + Precipitation + SpeciesDiversity + (1|Population),
                 data = biomass)
(R2_noIA <- partR2(mod_noIA, partvars = c("Temperature", "Precipitation"), data = biomass))

# combine both
(R2_comb <- mergeR2(R2_full, R2_noIA))

partR2  

 Partitioning R2 (R-square) in mixed models

Description

R2, semi-partial (part) R2 for predictors and their combinations as well as inclusive R2, structure coefficients and beta weights for Gaussian, Poisson and binomial mixed models.

Usage

partR2(
  mod, 
  partvars = NULL, 
  data = NULL, 
  R2_type = "marginal", 
  max_level = NULL, 
  nboot = NULL, 
  CI = 0.95, 
  parallel = FALSE,
partR2

expct = "meanobs",
oire = TRUE,
partbatch = NULL,
allow_neg_r2 = FALSE
)

Arguments

mod  Fitted lme4 model (a merMod object).
partvars  Character vector specifying the predictors (fixed effects) for which to partition the R2. Can be main effects like c(“Var1”, “Var2”) and interactions (”Var1:Var2”). Predictors specified in partvars have to be named precisely like the terms in the formula to fit the model.
data  The data.frame used to fit the lme4 model. If not provided, partR2 will try to fetch it.
R2_type  “marginal” or ”conditional” R2. With ”marginal”, the variance explained by fixed effects is calculated. With ”conditional”, the variance explained by both fixed and random effects is calculated.
max_level  Level up to which shared semi-partial R2s are calculated. The number of sets for which to calculate R2 increases exponentially, i.e. for 10 variables 2^10 - 1 R2s can be calculated. If you are only interested in the unique but not the shared effects, use max_level = 1. If interested in unique effects and combinations of two terms, use max_level = 2 etc.
nboot  Number of parametric bootstrap iterations for confidence interval estimation (defaults to NULL, i.e. no bootstrapping). Larger numbers of bootstraps give a better asymptotic CI, but may be time-consuming. Bootstrapping can be switched on by setting nboot = 1000.
CI  Width of the required confidence interval between 0 and 1 (defaults to 0.95).
parallel  If TRUE, computation uses future within furrr::map which allows parallelisation. However, it is necessary to specify a plan before running partR2(). To see which options you have, check ?future::plan and have a look at our vignette for details. When running RStudio, usually plan(multisession, workers = 4) is a good choice, when you want to use 4 cores. To detect how many cores you have, use parallel::detectCores(). If no plan is specified, partR2 will simply run sequentially.
expct  A string specifying the method for estimating the expectation in Poisson models with log link and in Binomial models with logit link (in all other cases the argument is ignored). The only valid terms are ’meanobs’, ’latent’, ’none’ (and ’liability for binary and proportion data). With the default ’meanobs’, the expectation is estimated as the mean of the observations in the sample. With ’latent’, the expectation is estimated from estimates of the intercept and variances on the link scale. While this is a preferred solution, it is susceptible to the distribution of fixed effect covariates and gives appropriate results typically only when all covariances are centered to zero. With ’liability’ estimates follow formulae as presented in Nakagawa & Schielzeth (2010). With ’none’, R2 is calculated without distribution specific variance in the denominator.
Logical, defaults to TRUE. This argument allows the user to prevent the automatic fitting of an observation level random effect (by setting it to FALSE) in Poisson and binomial models. The OLRE is used to account for overdispersion.

List of character vectors with predictors that should be fitted and removed together. For example, `partbatch = list(batch1 = c("V1", "V2", "V3"), batch2 = c("V4", "V5", "V6"))` would calculate part R2 only for combinations of predictors which contain the variables V1, V2, V3 together or/and V4, V5, V6 together. This is useful when the number of potential subsets gets too large to be computationally practical, for example when dummy coding is used. See our vignette for details. This feature is still experimental and should be used with caution.

Calculating part R2 involves fitting two models, one with and one without the predictor of interest. In cases where the predictor has little association with the response, the resulting part R2 value can become negative. By default we set negative values to 0, but by setting this parameter to TRUE, R2 values can become negative.

Returns an object of class partR2 that is a list with the following elements:

- **call**: model call
- **R2_type**: Marginal or conditional R2
- **R2**: R2 and confidence intervals for full model and semi-partial R2 for predictors and their combinations
- **SC**: Structure coefficients and confidence intervals. SC are the correlation between a predictor and the predicted response.
- **IR2**: Inclusive R2. This is SC^2 * R2_full.
- **BW**: Standardised model estimates (beta weights) for fixed effects. Beta weights for Gaussian models are calculated as beta * sd(x)/sd(y), with beta being the estimated slope of a fixed effect for predictor x and response y. Beta weight for Non-Gaussian models are calculated as beta * sd(x). Beta weights for interactions or polynomial terms are not informative at the moment and we recommend users to standardise variables themselves before fitting the model and to look at the model estimates (Ests) instead of beta weights (BW) in the partR2 output. See vignette for details.
- **Ests**: Model estimates and confidence intervals.
- **R2_boot**: Parametric bootstrap samples for R2 for full model and partitions
- **SC_boot**: Parametric bootstrap samples for structure coefficients
- **IR2_boot**: Parametric bootstrap samples for inclusive R2 values
- **BW_boot**: Parametric bootstrap samples for beta weights
- **Ests_boot**: Parametric bootstrap samples for model estimates
- **partvars**: Predictors to partition
- **CI**: Coverage of the confidence interval as specified by the CI argument.
- **boot_warnings**: Potential warnings from estimating partial R2s during parametric bootstrapping
- **boot_message**: Potential messages from estimating partial R2s during parametric bootstrapping. Common are for example singularity messages in lme4.
References


Examples

```r
# scale data
biomass[] <- lapply(biomass, function(x) if (is.double(x)) scale(x) else x)

# Gaussian data
mod <- lmer(Biomass ~ Year + Temperature + Precipitation + SpeciesDiversity + (1 | Population), data = biomass)

# R2
(R2_1 <- partR2(mod))

# R2 with CI
(R2_2 <- partR2(mod, R2_type = "marginal", nboot = 15, CI = 0.95))

# Part (semi-partial) R2s with CIs
(mod <- lmer(Biomass ~ Year + Temperature + Precipitation + SpeciesDiversity + (1 | Population), data = biomass))

(R2_3 <- partR2(mod, partvars = c("SpeciesDiversity", "Temperature", "Precipitation"), R2_type = "marginal", nboot = 10, CI = 0.95))
```

print.partR2  

Print a partR2 object

Description

Displays the results a partR2 object (i.e. the result of a partR2 function call) in a nice form.

Usage

```r
## S3 method for class 'partR2'
print(x, round_to = 4, ...)  
```

Arguments

- `x`: partR2 object returned from one of the partR2 functions
- `round_to`: defaults to 4 (decimals)
- `...`: Additional arguments; none are used in this method.
**Value**

No return value, prints concise results of partR2 calculation.

---

**summary.partR2**

*Complete summary of a partR2 object*

**Description**

Displays extended results of partR2, including R2, part (semi-partial) R2, inclusive R2, structure coefficients and beta weights.

**Usage**

```r
## S3 method for class 'partR2'
summary(object, round_to = 4, ests = FALSE, ...)
```

**Arguments**

- `object` partR2 object returned from one of the partR2 functions
- `round_to` Defaults to 4 (decimals)
- `ests` Defaults to FALSE, if TRUE, also prints raw model estimates.
- `...` Additional arguments; not used at the moment

**Value**

No return value, prints extended summary of partR2 calculation.
Index

* datasets
  biomass, 3
  Grasshoppers, 4
  GuineaPigs, 5

* models
  forestplot, 3
  print.partR2, 9
  summary.partR2, 10

biomass, 3

forestplot, 2, 3

Grasshoppers, 4

GuineaPigs, 5

mergeR2, 5

partR2, 2, 6

partR2-package, 2

print.partR2, 9

summary.partR2, 2, 10