

Package ‘combat.enigma’

October 17, 2024

Type Package

Title Fit and Apply ComBat, LMM, or Prescaling Harmonization for ENIGMA and Other Multisite MRI Data

Version 1.1

Date 2024-10-17

Description Fit and apply ComBat, linear mixed-effects models (LMM), or prescaling to harmonize magnetic resonance imaging (MRI) data from different sites. Briefly, these methods remove differences between sites due to using different scanning devices, and LMM additionally tests linear hypotheses. As detailed in the manual, the original ComBat function was first modified for the harmonization of MRI data (Fortin et al. (2017) <[doi:10.1016/j.neuroimage.2017.11.024](https://doi.org/10.1016/j.neuroimage.2017.11.024)>) and then modified again to create separate functions for fitting and applying the harmonization and allow missing values and constant rows for its use within the Enhancing Neuro Imaging Genetics through Meta-Analysis (ENIGMA) Consortium (Radua et al. (2020) <[doi:10.1016/j.neuroimage.2017.11.024](https://doi.org/10.1016/j.neuroimage.2017.11.024)>); this package includes the latter version. LMM calls ‘‘lme’’ massively considering specific brain imaging details. Finally, prescaling is a good option for fMRI, where different devices can have varying units of measurement.

License Artistic License 2.0

Depends R (>= 2.10)

Imports car, caret, Matrix, nlme

LazyData true

NeedsCompilation no

Author Joaquim Radua [aut, cre] (<<https://orcid.org/0000-0003-1240-5438>>)

Maintainer Joaquim Radua <quimradua@gmail.com>

Repository CRAN

Date/Publication 2024-10-17 14:00:02 UTC

Contents

combat_example	2
----------------	---

combat_fit	2
lmm_fit	5
prescale_fit	7

Index	9
--------------	----------

combat_example	<i>Simulated MRI data for combat_fit/apply</i>
----------------	--

Description

Simulated MRI data (FreeSurfer subcortical volumes) to provide an example for [combat_fit](#) and [combat_apply](#).

Author(s)

Joaquim Radua

See Also

[combat_fit](#) and [combat_apply](#)

combat_fit	<i>Fit and apply ComBat harmonization</i>
------------	---

Description

Fit and apply ComBat to harmonize magnetic resonance imaging (MRI) data from different sites. Briefly, ComBat is a batch adjustment method that removes additive and multiplicative differences between sites due to the use of different scanning devices. `combat_fit` fits the ComBat model, while `combat_apply` applies the harmonization using the model. As detailed below, the original `combat` function from the `sva` package was first modified by Jean-Philippe Fortin for the harmonization of MRI data and then modified by Joaquim Radua to create separate functions for fitting and applying the harmonization and allow missing values and constant rows for its use within the Enhancing Neuro Imaging Genetics through Meta-Analysis (ENIGMA) Consortium. Finding the effects of the site in one sample and then removing them from another sample is interesting in several scenarios, such as case-control studies (where the effects of the site are better found in controls) or in machine-learning studies (where the effects of the site must be found in the training sample).

Usage

```
combat_fit(dat, site, cov = NULL, n.min = 10, impute_missing_cov = FALSE,
           prescaling = FALSE, impute_empty_sites = FALSE, eb = TRUE, verbose = TRUE)
combat_apply(combat_parameters, dat, site, cov = NULL, verbose = TRUE)
```

Arguments

dat	matrix or data.frame with the MRI data (e.g., ROIs, voxels, vertexes, etc.).
site	factor specifying the site of each individual.
cov	matrix or data.frame with the covariates.
combat_parameters	a list of combat parameters returned by <code>combat_fit</code> .
n.min	(optional) number specifying the minimum size of a site to be analyzed.
impute_missing_cov	(optional) logical, whether to impute missing covariates.
prescaling	(optional) logical, whether to prescale the sites' data before conducting ComBat. See 'Details'.
impute_empty_sites	(optional) logical, whether to impute data when all participants of some sites have missing data for some ROIs or voxels. See 'Details'.
eb	(optional) logical, whether to use empirical Bayes.
verbose	(optional) logical, whether to print some messages during execution.

Details

The original ComBat function in the `sva` package was first modified by Jean-Philippe Fortin for the harmonization of MRI data and modified again by Joaquim Radua to create separate functions for fitting and applying the harmonization, allowing missings and constant rows and minor changes in the arguments of the functions to facilitate their use. The current version is thus based on the one described in "Increased power by harmonizing structural MRI site differences with the ComBat batch adjustment method in ENIGMA" below, but please also acknowledge the previous versions. In situations where the use of ComBat is problematic, such as when all participants of some sites have missing data for some ROIs or voxels, a linear mixed effects model (LMM) may be preferable; please see [lmm_fit](#). Finally, setting `prescaling` equal to `TRUE` may be especially beneficial when the sites use different scales, as it may be easily the case in fMRI mega-analyses (see "Neural correlates of human fear conditioning and sources of variability: A mega-analysis and normative modeling study of fMRI data from 2,199 individuals").

Value

`combat_fit` returns a list of ComBat parameters for `combat_apply`; `combat_apply` returns the list of parameters plus the harmonized data (item `dat.combat`)

Author(s)

Joaquim Radua, based on the previous work of others (see Details)

References

Fortin JP, Cullen N, Sheline YI, Taylor WD, Aselcioglu I, Cook PA, Adams P, Cooper C, Fava M, McGrath PJ, McInnis M, Phillips ML, Trivedi MH, Weissman MM, Shinohara RT. (2017) Harmonization of cortical thickness measurements across scanners and sites. *Neuroimage*, **167**, 104–120, doi:10.1016/j.neuroimage.2017.11.024.

Radua J, Vieta E, Shinohara R, Kochunov P, Quide Y, Green MJ, Weickert CS, Weickert T, Bruggemann J, Kircher T, Nenadic I, Cairns MJ, Seal M, Schall U, Henskens F, Fullerton JM, Mowry B, Pantelis C, Lenroot R, Croypley V, Loughland C, Scott R, Wolf D, Satterthwaite TD, Tan Y, Sim K, Piras F, Spalletta G, Banaj N, Pomarol-Clotet E, Solanes A, Albajes-Eizagirre A, Canales-Rodriguez EJ, Sarro S, Di Giorgio A, Bertolino A, Stablein M, Oertel V, Knochel C, Borgwardt S, du Plessis S, Yun JY, Kwon JS, Dannlowski U, Hahn T, Grotegerd D, Alloza C, Arango C, Janssen J, Diaz-Caneja C, Jiang W, Calhoun V, Ehrlich S, Yang K, Cascella NG, Takayanagi Y, Sawa A, Tomyshev A, Lebedeva I, Kaleda V, Kirschner M, Hoschl C, Tomecek D, Skoch A, van Amelsvoort T, Bakker G, James A, Preda A, Weideman A, Stein DJ, Howells F, Uhlmann A, Temmingh H, Lopez-Jaramillo C, Diaz-Zuluaga A, Fortea L, Martinez-Heras E, Solana E, Llufrui S, Jahanshad N, Thompson P, Turner J, van Erp T; ENIGMA Consortium collaborators. (2020) Increased power by harmonizing structural MRI site differences with the ComBat batch adjustment method in ENIGMA. *Neuroimage*, **218**, 116956, doi:10.1016/j.neuroimage.2020.116956.

Neural correlates of human fear conditioning and sources of variability: A mega-analysis and normative modeling study of fMRI data from 2,199 individuals, submitted.

See Also

[prescale_fit](#) and [lmm_fit](#)

Examples

```
raw_mri = combat_example[,6:19]
site = factor(combat_example$site)

# Fit and apply ComBat to harmonize mri data across sites
mod = as.matrix(combat_example[,c("disorder", "age", "sex")])
combat = combat_fit(raw_mri, site, mod)
harmonized_mri = combat_apply(combat, raw_mri, site, mod)$dat.combat

# Run ANOVAs to check the effects of site
Out = NULL
for (j in 1:ncol(raw_mri)) {
  raw_mri_anova = anova(lm(raw_mri[,j] ~ mod), lm(raw_mri[,j] ~ mod + site))
  harmonized_mri_anova = anova(lm(harmonized_mri[,j] ~ mod), lm(harmonized_mri[,j] ~ mod + site))
  Out = rbind(Out, data.frame(
    roi = colnames(raw_mri)[j],
    raw_mri.F = round(raw_mri_anova$F[2], 1),
    raw_mri.P = round(raw_mri_anova$`Pr(>F)`[2], 3),
    harmonized_mri.F = round(harmonized_mri_anova$F[2], 1),
    harmonized_mri.P = round(harmonized_mri_anova$`Pr(>F)`[2], 3)
  ))
}
Out[,-1] = apply(Out[,-1], 2, function (x) {ifelse(x == 0, "<0.001", x)})
Out
```

Imm_fit

*Fit LMM harmonization and obtain model coefficients***Description**

Fit linear mixed-effects models (LMM) to test magnetic resonance imaging (MRI) data from different sites. Briefly, this function calls `lme` (from the `nlme` package) massively to fit LMM for each ROI/voxel/vertex with the variables in `cov` as fixed-effects factors and `site` as a random intercept. Importantly, it also addresses specific brain imaging details (e.g., prescaling fMRI measures if needed or accounting for varying collinearity due to differing brain coverage; see "Neural correlates of human fear conditioning and sources of variability: A mega-analysis and normative modeling study of fMRI data from 2,199 individuals").

Usage

```
Imm_fit(dat, site, cov = NULL, n.min = 10, impute_missing_cov = FALSE,
        prescaling = FALSE, lin.hyp = NULL, verbose = TRUE)
```

Arguments

<code>dat</code>	matrix or data.frame with the MRI data (e.g., ROIs, voxels, vertexes, etc.).
<code>site</code>	factor specifying the site of each individual.
<code>cov</code>	matrix or data.frame with the fixed-effects covariates.
<code>n.min</code>	(optional) number specifying the minimum size of a site to be analyzed.
<code>impute_missing_cov</code>	(optional) logical, whether to impute missing covariates.
<code>prescaling</code>	(optional) logical, whether to prescale the sites' data before conducting ComBat. See 'Details'.
<code>lin.hyp</code>	A hypothesis vector or matrix giving linear combinations of coefficients by rows as described in linearHypothesis (from the <code>car</code> package).
<code>verbose</code>	(optional) logical, whether to print some messages during execution.

Details

In several situations, the use of ComBat is problematic, such as when all participants of some sites have missing data for some ROIs or voxels. In these cases, a linear mixed effects model (LMM) may be preferable. On another note, setting `prescaling` equal to `TRUE` may be especially beneficial when the sites use different scales, as may be easily the case in fMRI mega-analyses.

Value

A list of parameters plus the results of the LMM.

Author(s)

Joaquim Radua

References

Neural correlates of human fear conditioning and sources of variability: A mega-analysis and normative modeling study of fMRI data from 2,199 individuals, submitted.

See Also

[combat_fit](#), [prescale_fit](#), and [linearHypothesis](#)

Examples

```
raw_mri = combat_example[,6:19]
site = factor(combat_example$site)
mod = as.matrix(combat_example[,c("disorder", "age", "sex")])

# Estimate the effects of disorder with simple linear models
# WITHOUT harmonizing MRI data across sites
Out_raw = t(apply(raw_mri, 2, function(y) {
  m = summary(lm(y ~ mod, data = combat_example))
  coef(m)[2,c(1,3,4)]
}))

# Estimate the effects of disorder with simple linear models
# AFTER ComBat harmonizing MRI data across sites
combat = combat_fit(raw_mri, site, mod)
harmonized_mri = combat_apply(combat, raw_mri, site, mod)$dat.combat
Out_combat = t(apply(harmonized_mri, 2, function(y) {
  m = summary(lm(y ~ mod, data = combat_example))
  coef(m)[2,c(1,3,4)]
}))

# Estimate the effects of disorder with LMM harmonizing MRI data across sites
lmm = lmm_fit(raw_mri, site, mod)
Out_lmm = data.frame(
  b = lmm$b_map[[2]],
  t = lmm$t_map[[2]],
  p = 2 * pnorm(-abs(lmm$z_map[[2]]))
)
rownames(Out_lmm) = colnames(raw_mri)

# Results without harmonizing, with combat_fit and with lmm_fit:
cat("\nRaw results:\n")
print(round(Out_raw, 3))
cat("\nComBat results:\n")
print(round(Out_combat, 3))
cat("\nLMM results:\n")
print(round(Out_lmm, 3))

# Correlations between the three methods:
cat("\nCorrelation in coefficients:\n")
print(round(cor(cbind(Out_raw[,1], Out_combat[,1], Out_lmm[,1])), 3))
cat("\nCorrelation in p-values:\n")
print(round(cor(cbind(Out_raw[,3], Out_combat[,3], Out_lmm[,3])), 3))
```

`prescale_fit`*Fit and apply brain imaging prescaling*

Description

Fit and apply a prescaling of magnetic resonance imaging (MRI) data from different sites, especially relevant for fMRI, where different devices can have varying units of measurement.

Usage

```
prescale_fit(dat, site, cov = NULL, n.min = 10, impute_missing_cov = FALSE,
             verbose = TRUE)
prescale_apply(combat_parameters, dat, site, cov = NULL, verbose = TRUE)
```

Arguments

<code>dat</code>	matrix or data.frame with the MRI data (e.g., ROIs, voxels, vertexes, etc.).
<code>site</code>	factor specifying the site of each individual.
<code>cov</code>	matrix or data.frame with the covariates.
<code>combat_parameters</code>	a list of combat parameters returned by <code>prescale_fit</code> .
<code>n.min</code>	(optional) number specifying the minimum size of a site to be analyzed.
<code>impute_missing_cov</code>	(optional) logical, whether to impute missing covariates.
<code>verbose</code>	(optional) logical, whether to print some messages during execution.

Details

`prescale_fit` function finds a prescaling parameter for each site so that, after calling `prescale_apply`, the voxelwise-median standard deviation after removing the effects of covariates is 1 (in the training data, but it might not be the case in new data).

Value

`prescale_fit` returns a list of parameters for `prescale_apply`; `prescale_apply` returns the list of parameters plus the prescaled data (item `dat.combat`)

Author(s)

Joaquim Radua

References

Neural correlates of human fear conditioning and sources of variability: A mega-analysis and normative modeling study of fMRI data from 2,199 individuals, submitted.

See Also

[combat_fit](#), and [lmm_fit](#)

Examples

```
raw_mri = combat_example[,6:19]
site = factor(combat_example$site)

# Fit and apply prescale to prescale mri data across sites
mod = as.matrix(combat_example[,c("disorder", "age", "sex")])
prescaling = prescale_fit(raw_mri, site, mod)
prescaled_mri = prescale_apply(prescaling, raw_mri, site, mod)$dat.combat
```


Index

* **enigma**

combat_fit, 2
lmm_fit, 5
prescale_fit, 7

* **harmonization**

combat_fit, 2
lmm_fit, 5
prescale_fit, 7

* **mri**

combat_fit, 2
lmm_fit, 5
prescale_fit, 7

combat_apply, 2
combat_apply (combat_fit), 2
combat_example, 2
combat_fit, 2, 2, 6, 8

linearHypothesis, 5, 6
lme, 5
lmm_fit, 3, 4, 5, 8

prescale_apply (prescale_fit), 7
prescale_fit, 4, 6, 7