Package 'ZIBR'

January 20, 2025

Type Package

Title A Zero-Inflated Beta Random Effect Model

Version 1.0.2

Date 2023-10-16

Description A two-part zero-inflated Beta regression model with random effects (ZIBR) for testing the association between microbial abundance and clinical covariates for longitudinal microbiome data. Eric Z. Chen and Hongzhe Li (2016) <doi:10.1093/bioinformatics/btw308>.

License MIT + file LICENSE

LazyData TRUE

RoxygenNote 7.2.3

Encoding UTF-8

Suggests betareg, dplyr, lme4 (>= 1.1-34), nlme, knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

Depends R (>= 2.10), statmod

VignetteBuilder knitr

URL https://github.com/PennChopMicrobiomeProgram/ZIBR

BugReports https://github.com/PennChopMicrobiomeProgram/ZIBR/issues

NeedsCompilation no

Author Eric Zhang Chen [aut, cph], Charlie Bushman [cre]

Maintainer Charlie Bushman <ctbushman@gmail.com>

Repository CRAN

Date/Publication 2023-10-17 09:50:08 UTC

Contents

`fit_beta_random_effect`

Fit beta random effect

Description

Fit beta random effect

Usage

```
fit_beta_random_effect(
  Z = Z,
  Y = Y,
  subject.ind = subject.ind,
  time.ind = time.ind,
  quad.n = 30,
  verbose = FALSE
)
```

Arguments

`Z`	FILL
`Y`	FILL
`subject.ind`	the subject index
`time.ind`	the time index
`quad.n`	number of points in gaussian quadrature
`verbose`	a boolean to enable more output

Value

a named list

- est.table
- s2.est
- v.est

`fit_logistic_random_effect` *Fit logisitic random effect*

Description

Fit logisitic random effect

Usage

```
fit_logistic_random_effect(
  X = X,
  Y = Y,
  subject.ind = subject.ind,
  time.ind = time.ind,
  quad.n = 30,
  verbose = FALSE
)
```

Arguments

`X`	FILL
`Y`	FILL
`subject.ind`	the subject index
`time.ind`	the time index
`quad.n`	number of points in gaussian quadrature
`verbose`	a boolean to enable more output

Value

a named list

- est.table
- s1.est

`fit_zero_inflated_beta_random_effect` *Fit zero inflated beta random effect*

Description

Fit zero inflated beta random effect

Usage

```
fit_zero_inflated_beta_random_effect(
  X = X,
  Z = Z,
  Y = Y,
  subject_ind = subject_ind,
  time_ind = time_ind,
  component_wise_test = TRUE,
  joint_test = TRUE,
  quad_n = 30,
  verbose = FALSE
)
```

Arguments

X	FILL
Z	FILL
Y	FILL
subject_ind	the subject index
time_ind	the time index
component_wise_test	boolean to run component-wise test
joint_test	boolean to run joint test
quad_n	number of points in gaussian quadrature
verbose	a boolean to enable more output

Value

a named list

- logistic_est_table
- logistic_s1_est
- beta_est_table
- beta_s2_est
- beta_v_est
- loglikelihood
- joint_p

ibd *Longitudinal human microbiome data*

Description

A dataset containing the bacterial abundance and clinical information from a longitudinal human microbiome study

Usage

```
ibd
```

Format

A data frame with 236 rows and 5 variables:

Sample Sample IDs

Subject Subject IDs

Time Time points

Treatment Treatment, 0 for antiTNF, 1 for EEN

Abundance Abundance for Eubacterium ...

References

Lewis and Chen et al. (2016) Cell Host & Microbe 18 (4), 489-500

simulate_beta_random_effect_data *Simulate beta data*

Description

Simulate beta data

Usage

```
simulate_beta_random_effect_data(
  subject_n = 50,
  time_n = 5,
  v = 2,
  beta = as.matrix(c(-0.5, -0.5, 0.5)),
  Z = NA,
  s2 = 1,
  sim_seed = 100
)
```

Arguments

subject_n	the number of subjects
time_n	the number of time points
v	FILL
beta	FILL
Z	FILL
s2	FILL
sim_seed	the random seed with which to simulate the data

Value

a named list

- Y
- Z
- c
- u
- v
- beta
- s2
- subject_ind
- time_ind

---

simulate_logistic_data

Simulate logistic data

Description

Simulate logistic data

Usage

```
simulate_logistic_data(
  subject_n = 50,
  time_n = 5,
  alpha = as.matrix(c(0, 0.5, -1)),
  s1 = 0.5,
  sim_seed = 100
)
```

Arguments

subject_n	the number of subjects
time_n	the number of time points
alpha	FILL
s1	FILL
sim_seed	the random seed with which to simulate the data

Value

a named list

- X
- Y
- b
- subject_ind
- time_ind

---

simulate_zero_inflated_beta_random_effect_data

Simulate data according to zero-inflated beta random effects model

---

Description

Simulate data according to zero-inflated beta random effects model

Usage

```
simulate_zero_inflated_beta_random_effect_data(
  subject_n = 50,
  time_n = 5,
  v = 2,
  alpha = as.matrix(c(0, 0.5, -1)),
  beta = as.matrix(c(-0.5, -0.5, 0.5)),
  X = NA,
  Z = NA,
  s1 = 0.2,
  s2 = 0.2,
  sim_seed = 100
)
```

Arguments

subject_n	number of subjects
time_n	number of time points for each subject
v	the dispersion parameter in beta component
alpha	the coefficients in logistic component
beta	the coefficients in beta component
X	the covariates in logistic component
Z	the covariates in beta component
s1	the stardard deviation of random effect in logistic component
s2	the stardard deviation of random effect in beta component
sim_seed	the random seed

Value

a named list

- Y the bacterial abundance generated from the model
- X the covariates in logistic component
- Z the covariates in beta component
- alpha the coefficients in logistic component
- beta the coefficients in beta component
- s1 the stardard deviation of random effect in logistic component
- s2 the stardard deviation of random effect in beta component
- subject_ind the IDs for each subject
- time_ind time points

Examples

```
simulate_zero_inflated_beta_random_effect_data(
  subject_n = 100, time_n = 5,
  X = as.matrix(c(rep(0, 50 * 5), rep(1, 50 * 5))),
  alpha = as.matrix(c(-0.5, 1)),
  beta = as.matrix(c(-0.5, 0.5)),
  s1 = 1, s2 = 0.8,
  v = 5,
  sim_seed = 100
)
```

zibr *Fit zero-inflated beta regression with random effects*

Description

Fit zero-inflated beta regression with random effects

Usage

```
zibr(
  logistic_cov,
  beta_cov,
  Y,
  subject_ind,
  time_ind,
  component_wise_test = TRUE,
  quad_n = 30,
  verbose = FALSE
)
```

Arguments

logistic_cov	the covariates in logistic component
beta_cov	the covariates in beta component
Y	the response variable in the regression model
subject_ind	the variable for subject IDs
time_ind	the variable for time points
component_wise_test	whether to perform component wise test. If true, ZIBR will calculate p-values for logistic and beta component respectively.
quad_n	Gaussian quadrature points
verbose	print the fitting process

Value

a named list

- logistic_est_table - the estimated coefficients for logistic component.
- logistic_s1_est - the estimated standard deviation for the random effect in the logistic component.
- beta_est_table - the estimated coefficients for logistic component.
- beta_s2_est - the estimated standard deviation for the random effect in the beta component.
- beta_v_est - the estimated dispersion parameter in the beta component.
- loglikelihood - the log likelihood of fitting ZIBR model on the data.
- joint_p - the p-values for jointly testing each covariate in both logistic and beta component.

Examples

```
## simulate some data
sim <- simulate_zero_inflated_beta_random_effect_data(
  subject_n = 100, time_n = 5,
  X = as.matrix(c(rep(0, 50 * 5), rep(1, 50 * 5))),
  Z = as.matrix(c(rep(0, 50 * 5), rep(1, 50 * 5))),
  alpha = as.matrix(c(-0.5, 1)),
  beta = as.matrix(c(-0.5, 0.5)),
  s1 = 1, s2 = 0.8,
  v = 5,
  sim_seed = 100
)

## run zibr on the simulated data
zibr_fit <- zibr(
  logistic_cov = sim$X, beta_cov = sim$Z, Y = sim$Y,
  subject_ind = sim$subject_ind, time_ind = sim$time_ind
)

zibr_fit
```

Index