Package 'fastFMM'

February 2, 2024

Type Package

Title Fast Functional Mixed Models using Fast Univariate Inference

Version 0.2.0

Date 2024-02-02

Description Implementation of the fast univariate inference approach (Cui et al. (2022) <doi:10.1080/10618600.2021.1950006>, Loewinger et al. (2023) <doi:10.1101/2023.11.06.56589
ting functional mixed models.

License GPL (>= 3)

Encoding UTF-8

Repository CRAN

Imports lme4, parallel, cAIC4, magrittr, dplyr, mgcv, MASS, lsei, refund, stringr, Matrix, mvtnorm, progress, ggplot2, gridExtra, Rfast, lmeresampler, stats, methods

RoxygenNote 7.2.3

URL https://github.com/gloewing/fastFMM

BugReports https://github.com/gloewing/fastFMM/issues

VignetteBuilder knitr

Suggests knitr, rmarkdown, spelling

Language en-US

NeedsCompilation no

Author Erjia Cui [aut, cre], Gabriel Loewinger [aut]

Maintainer Erjia Cui <ecui@umn.edu>

Date/Publication 2024-02-02 17:30:02 UTC

R topics documented:

fui *Fast Univariate Inference for Longitudinal Functional Models*

Description

Fit a function-on-scalar regression model for longitudinal functional outcomes and scalar predictors using the Fast Univariate Inference (FUI) approach (Cui et al. 2022).

Usage

```
fui(
  formula,
  data,
  family = "gaussian",
  var = TRUE,
  analytic = TRUE,
  parallel = FALSE,
  silent = FALSE,
  argvals = NULL,
  nknots_min = NULL,
  nknots_min_cov = 35,
  smooth_method = "GCV.Cp",
  splines = "tp",
  design_mat = FALSE,
  residuals = FALSE,
  G_return = FALSE,
  num_boots = 500,
  boot_type = NULL,
  seed = 1,
  subj_ID = NULL,
  num_cores = 1,
  caic = FALSE,
  REs = FALSE,
  non_neg = 0,
  MoM = 2
)
```

Arguments

formula	Two-sided formula object in lme4 formula syntax. The difference is that the response need to be specified as a matrix instead of a vector. Each column of the matrix represents one location of the longitudinal functional observations on the domain.
data	A data frame containing all variables in formula
family	GLM family of the response. Defaults to gaussian.

var	Logical, indicating whether to calculate and return variance of the coefficient estimates. Defaults to TRUE.
analytic	Logical, indicating whether to use the analytic inference approach or bootstrap. Defaults to TRUE.
parallel	Logical, indicating whether to do parallel computing. Defaults to FALSE.
silent	Logical, indicating whether to show descriptions of each step. Defaults to FALSE.
argvals	A vector containing locations of observations on the functional domain. If not specified, a regular grid across the range of the domain is assumed. Currently only supported for bootstrap (analytic=FALSE).
nknots_min	Minimal number of knots in the penalized smoothing for the regression coefficients. Defaults to NULL, which then uses L/2 where L is the dimension of the functional domain.
nknots_min_cov	Minimal number of knots in the penalized smoothing for the covariance matrices. Defaults to 35.
smooth_method	How to select smoothing parameter in step 2. Defaults to "GCV.Cp"
splines	Spline type used for penalized splines smoothing. We use the same syntax as the mgcv package. Defaults to "tp"
design_mat	Logical, indicating whether to return the design matrix. Defaults to FALSE
residuals	Logical, indicating whether to save residuals from unsmoothed LME. Defaults to FALSE.
G_return	Logical, indicating whether to return (smoothed and trimmed) G = Cov(u(s_t), u(s_l)). Defaults to FALSE.
num_boots	Number of samples when using bootstrap inference. Defaults to 500.
boot_type	Bootstrap type (character): "cluster", "case", "wild", "reb", "residual", "parametric", "semiparametric". NULL defaults to "cluster" for non-gaussian responses and "wild" for gaussian responses. For small cluster (n<=10) gaussian responses, defaults to "reb"
seed	Numeric value used to make sure bootstrap replicate (draws) are correlated across functional domains for certain bootstrap approach
subj_ID	Name of the variable that contains subject ID.
num_cores	Number of cores for parallelization. Defaults to 1.
caic	Logical, indicating whether to calculate cAIC. Defaults to FALSE.
REs	Logical, indicating whether to return random effect estimates. Defaults to FALSE.
non_neg	0 - no non-negativity constrains, 1 - non-negativity constraints on every coefficient for variance, 2 - non-negativity on average of coefficents for 1 variance term. Defaults to 0.
MoM	Method of moments estimator. Default to 2. 1 should only be used for extremely large datasets.

Details

The FUI approach comprises of three steps:

1. Fit a univariate mixed model at each location of the functional domain, and obtain raw estimates from massive models;
2. Smooth the raw estimates along the functional domain;
3. Obtain the pointwise and joint confidence bands using an analytic approach for Gaussian data or Bootstrap for general distributions.

For more information on each step, please refer to the FUI paper by Cui et al. (2022).

Value

A list containing:

`betaHat`	Estimated functional fixed effects
`argvals`	Location of the observations
`betaHat.var`	Variance estimates of the functional fixed effects (if specified)
`qn`	critical values used to construct joint CI
`...`	...

Author(s)

Erjia Cui <ecui@umn.edu>, Gabriel Loewinger <gloewinger@gmail.com>

References

Cui, E., Leroux, A., Smirnova, E., Crainiceanu, C. (2022). Fast Univariate Inference for Longitudinal Functional Models. *Journal of Computational and Graphical Statistics*, 31(1), 219-230.

Examples

```
library(refund)

## random intercept only
set.seed(1)
DTI_use <- DTI[DTI$ID %in% sample(DTI$ID, 10),]
fit_dti <- fui(cca ~ case + visit + sex + (1 | ID),
                data = DTI_use)
```

`plot_fui` *Default FUI plotting*

Description

Take a fitted `fui` object produced by `fastFMM::fui()` and plot the point estimates of fixed effects. When variance was calculated, the plot function also returns 95% pointwise and joint confidence intervals.

Usage

```
plot_fui(
  fuiobj,
  num_row = NULL,
  xlab = "Functional Domain",
  title_names = NULL,
  ylim = NULL,
  align_x = NULL,
  x_rescale = 1,
  y_val_lim = 1.1,
  y_scal_orig = 0.05,
  return = FALSE
)
```

Arguments

`fuiobj`	A object returned from the `fui` function
`num_row`	An integer that specifies the number of rows the plots will be displayed on. Defaults to p/2, where p is the number of predictors.
`xlab`	A string that specifies the x-axis title (i.e., for the functional domain). Defaults to "Functional Domain"
`title_names`	A vector of strings that has length equal to number of covariates (plus intercept if relevant). Allows one to change the titles of the plots. Defaults to NULL which uses the variable names in the dataframe for titles.
`ylim`	A 2-dimensional vector that specifies limits of the y-axis in plots.
`align_x`	A scalar: aligns the plot to a certain point on the functional domain and sets this as 0. This is particularly useful if the functional domain is time. Defaults to 0.
`x_rescale`	A scalar: rescales the x-axis of plots which is especially useful if time is the functional domain and one wishes to, for example, account for the sampling rate. Defaults to 1.
`y_val_lim`	A positive scalar that extends the y-axis by a factor for visual purposes. Defaults to 1.10. Typically does not require adjustment.
`y_scal_orig`	A positive scalar that determines how much to reduce the length of the y-axis on the bottom. Defaults to 0.05. Typically does not require adjustment.
`return`	Logical, indicating whether to return the data frame with the coefficient estimates and 95% confidence intervals (CIs). Defaults to `FALSE`.

Value

Plots of point estimates and CIs. If `return = TRUE`, also returns a list where each element is a data frame with the coefficient estimates and 95% confidence intervals (CIs).

Author(s)

Gabriel Loewinger <gloewinger@gmail.com>, Erjia Cui <ecui@umn.edu>

References

Cui, E., Leroux, A., Smirnova, E., Crainiceanu, C. (2022). Fast Univariate Inference for Longitudinal Functional Models. *Journal of Computational and Graphical Statistics*, 31(1), 219-230.

Examples

```
library(refund)
set.seed(1)
DTI_use <- DTI[DTI$ID %in% sample(DTI$ID, 6),]
fit_dti <- fui(formula = cca ~ case + visit + sex + (1 | ID),
               data = DTI_use, family = "gaussian", var = TRUE)
plot_fui(fit_dti)
```

Index