

Package ‘MIXFIM’

July 21, 2025

Type Package

Title Evaluation of the FIM in NLMEMs using MCMC

Version 1.1

Date 2018-11-16

Author Marie-Karelle Riviere-Jourdan and France Mentre <france.mentre@inserm.fr>

Maintainer Marie-Karelle Riviere-Jourdan <eldamjh@gmail.com>

Copyright All files are copyright Institut National de la Sante Et de la Recherche Medicale.

Description Evaluation and optimization of the Fisher Information Matrix in NonLinear Mixed Effect Models using Markov Chains Monte Carlo for continuous and discrete data.

License GPL-3

Depends R (>= 3.0.2), rstan (>= 2.7.0-1), mvtnorm (>= 1.0-2), ggplot2 (>= 1.0.1)

NeedsCompilation no

Repository CRAN

Date/Publication 2018-11-16 13:20:02 UTC

Contents

MIXFIM-package	2
fisher_evaluation	2
fisher_optimization	5
template_model	8

Index	10
--------------	-----------

MIXFIM-package

Evaluation of the FIM in NLMEMs using MCMC

Description

Evaluation and optimization of the Fisher Information Matrix in NonLinear Mixed Effect Models using Markov Chains Monte Carlo for continuous and discrete data.

Details

Package: MIXFIM
Type: Package
Version: 1.1
Date: 2018-11-16
License: GPL-3

Author(s)

Marie-Karelle Riviere-Jourdan <eldamjh@gmail.com> and France Mentre <france.mentre@inserm.fr>

References

Riviere, M-K., Ueckert, S. and Mentre, F. Evaluation of the Fisher information matrix in nonlinear mixed effect models using Markov Chains Monte Carlo.

fisher_evaluation

Evaluation of the Fisher Information Matrix in Nonlinear Mixed Effect Models using Markov Chains Monte Carlo

Description

fisher_evaluation is used to evaluate the Fisher information matrix for both continuous and discrete data in nonlinear mixed effect models using Markov Chains Monte Carlo.

Usage

```
fisher_evaluation(t, y_ini=1, model, model2, model3, params, dim_b,  
set_seed=TRUE, seed=42, n_samp, n_rep=1, n_iter, n_burn, CV=FALSE,  
plot_graph=0, L_boot=1000, nb_patients=1)
```

Arguments

t	Vector of sampling times (or doses).
y_ini	A possible value for the response y to initialize the MCMC process. The default value is set at 1 (which works for many types of outcomes: continuous, binary, ...).
model	Compiled STAN model describing the response model to sample in the conditional distribution of b given y.
model2	Compiled STAN model describing the response model for calculating the derivative of the log-likelihood with respect to each parameter.
model3	Compiled STAN model describing the response model to sample in the marginal distribution of the response y.
params	Vector of parameters given as follows: fixed effects, variances of the random effects, standard deviations of the residual errors (if continuous data).
dim_b	Number of random effects.
set_seed	Boolean indicating if the seed should be fixed. The default value is set at TRUE.
seed	Integer for the fixed seed. Used only if set_seed is TRUE. The default value is set at 42.
n_samp	Integer representing the number of Monte Carlo (MC) samples, (i.e. number of samples for the outcome y).
n_rep	Integer representing the number of repeated measures at the same time (or dose) for each patient. The default value is set at 1 (for continuous data).
n_iter	Integer representing the number of Markov Chains Monte Carlo (MCMC) samples.
n_burn	Integer representing the number of burn-in samples for MCMC.
CV	Boolean indicating if some convergence information (variance of the determinant, mean of b, mean log-likelihood, ...) should be returned. The default value is set at FALSE.
plot_graph	An integer with value 0 (no graph should be plotted), 1 (graph of the determinant of the FIM), 2 (graph of the determinant of the FIM with confidence intervals assuming normal distribution), 3 (graph of the determinant of the FIM with bootstrap confidence intervals) or 4 (graph of the determinant of the FIM with both bootstrap confidence intervals and confidence intervals assuming normal distribution). The default value is set at 0.
L_boot	Number of samples for bootstrap estimation of the confidence intervals of the normalized determinant of the FIM. This argument is used/required only if plot_graph = 3 or 4. The default value is set at 1000.
nb_patients	Number of patients with the same elementary design for which the FIM is evaluated. The default value is set at 1.

Value

An list is returned, composed of the following variables:

FIM	Expected Fisher information matrix (FIM). Of note, the FIM is an individual FIM and is calculated for nb_patients patients.
FIM_covar	Variance-covariance matrix of the FIM. (Of note, its dimension is of size 4 as the FIM is in dimension 2.)
inv_FIM	Inverse of the FIM.
RSE	Relative standard errors (square root of the diagonal elements of the inverse of the FIM).
RSE_inf_boot	Vector containing the lower bound of the bootstrap confidence interval of the RSEs.
RSE_sup_boot	Vector containing the upper bound of the bootstrap confidence interval of the RSEs.
det_norm_FIM	Normalized determinant of the FIM.
det_IC_normal	Vector containing the lower and upper bound of the confidence interval of the normalized determinant of the FIM assuming normal distribution.
det_IC_boot	Vector containing the lower and upper bound of the bootstrap confidence interval of the normalized determinant of the FIM.
If CV=TRUE:	
mean_dloglik1	Mean of the partial derivatives of the log-likelihood according to the first MCMC sample and MC sample. Should be equal approximately to 0.
mean_dloglik2	Mean of the partial derivatives of the log-likelihood according to the second MCMC sample and MC sample. Should be equal approximately to 0.
var_dloglik1	Variance of the partial derivatives of the log-likelihood according to the first MCMC sample and MC sample.
var_dloglik2	Variance of the partial derivatives of the log-likelihood according to the second MCMC sample and MC sample.
mean_b	Mean of the samples in the conditionnal distribution of b given y. Should be equal approximately to 0.
mat_A_k1	Vector containing for each value sampled of the response y, the estimation of the integral of the partial derivatives of the log-likelihood over the random effects according to the first MCMC sample of the random effects b given y.
mat_A_k2	Vector containing for each value sampled of the response y, the estimation of the integral of the partial derivatives of the log-likelihood over the random effects according to the second MCMC sample of the random effects b given y.

In addition, plot_graph enables to plot a graph of the normalized determinant of the FIM with normal and bootstrap confidence intervals in function of the number of MC samples.

Author(s)

Marie-Karelle Riviere-Jourdan <eldamjh@gmail.com>

References

Riviere, M-K., Ueckert, S. and Mentre, F. Evaluation of the Fisher information matrix in nonlinear mixed effect models using Markov Chains Monte Carlo.

Examples

```
#####
# PLEASE UNCOMMENT EXAMPLE #
#####
#times = c(0.5,1,2,6,24,36,72,120)
#params = c(1,8,0.15,0.6,0.02,0.07,0.1)

# Files can be found in external data
#model = stan_model("model_b_given_y.stan")
#model2 = stan_model("model_derivatives.stan")
#model3 = stan_model("model_y.stan")

#model_Warfarin = fisher_evaluation(t=times, y_ini=0.5, model=model,
#model2=model2, model3=model3, params=params, dim_b=3, set_seed=TRUE, seed=42,
#n_samp=1000, n_rep=1, n_iter=200, n_burn=500, CV=TRUE, plot_graph=4,
#nb_patients=32)
#model_Warfarin
```

fisher_optimization	<i>Optimization of the sampling times (or doses) in Nonlinear Mixed Effect Models using Markov Chains Monte Carlo</i>
---------------------	---

Description

fisher_optimization is used to optimize the sampling times (or doses), through maximization of the determinant of the Fisher Information Matrix, for both continuous and discrete data in nonlinear mixed effect models using Markov Chains Monte Carlo.

Usage

```
fisher_optimization(nb_t, set_t, y_ini, model, model2, model3, params, dim_b,
set_seed=TRUE, seed=42, step_mc, n_samp_min=30, n_samp_max, n_rep=1, n_iter,
n_burn, L_boot=1000, plot_graph=TRUE, nb_patients=1, det_graph=FALSE)
```

Arguments

nb_t	Number of sampling times (or doses) to select.
set_t	A vector containing the set of possible sampling times in which to select the nb_t optimal ones.
y_ini	A possible value for the response y to initialize the MCMC process.
model	Compiled STAN model describing the response model to sample in the conditional distribution of b given y.
model2	Compiled STAN model describing the response model for calculating the derivative of the log-likelihood with respect to each parameter.
model3	Compiled STAN model describing the response model to sample in the marginal distribution of the response y.

params	Vector of parameters given as follows: fixed effects, variances of the random effects, standard deviations of the residual errors (if continuous data).
dim_b	Number of random effects.
set_seed	Boolean indicating if the seed should be fixed. The default value is set at TRUE.
seed	Integer for the fixed seed. Used only if set_seed is TRUE. The default value is set at 42.
step_mc	Number of Monte Carlo (MC) samples after which the remaining combinations of sampling times are re-evaluated for optimum determination.
n_samp_min	Integer representing the minimum number of MC samples to perform at the first step to evaluate all combinations of sampling times (or doses).
n_samp_max	Integer representing the maximum number of MC samples to perform (for one combination of sampling times). Either the algorithm will stop for finding a unique combination of optimal sampling times before reaching n_samp_max or for reaching it. In the second case, the sampling times corresponding to the highest determinant of the FIM will be selected as the optimum.
n_rep	Integer representing the number of repeated measures at the same time (or dose) for each patient. The default value is set at 1 (for continuous data).
n_iter	Integer representing the number of Markov Chains Monte Carlo (MCMC) samples.
n_burn	Integer representing the number of burn-in samples for MCMC.
L_boot	Number of samples for bootstrap estimation of the confidence intervals of the determinant of the FIM. The default value is set at 1000.
plot_graph	A boolean indicating if at each step (every step_mc), a graph with the determinant of the FIM and its bootstrap confidence intervals for the remaining combinations of sampling times should be plotted and saved. The default value is set at TRUE.
nb_patients	Number of patients with the same elementary design for which the FIM and its determinant are evaluated. The default value is set at 1.
det_graph	A boolean indicating if the normalized determinant of the FIM for the selected optimal design should be plotted in function of the number of MC samples. The default value is set at FALSE.

Details

The algorithm will first test all combinations of nb_t sampling times (or doses) in ens_t on at least n_samp_min MC samples. Then and after each step (every step_mc), only combinations of sampling times for which the upper bound of the bootstrap confidence interval is above the maximum estimated determinant of the FIM will be selected for the next step. The algorithm will stop either for finding a unique combination of optimal sampling times before reaching n_samp_max or for reaching it. In the second case, the combination of sampling times corresponding to the highest determinant of the FIM will be selected as the optimum. However, in this case the graph (plot_graph=TRUE) could help to distinguish the remaining combinations of sampling times.

Value

An list is returned, composed of the following variables:

opt_t	Optimal protocol, i.e. sampling times (or doses).
FIM_opt_t	Expected Fisher information matrix (FIM) for the optimal protocol. Of note, the FIM is an individual FIM and is calculated for nb_patients patients.
FIM_covar_opt_t	Variance-covariance matrix of the FIM for the optimal protocol. (Of note, its dimension is of size 4 as the FIM is in dimension 2.)
inv_FIM_opt_t	Inverse of the FIM for the optimal protocol.
RSE_opt_t	Relative standard errors for the optimal protocol (square root of the diagonal elements of the inverse of the FIM).
RSE_inf_boot_opt_t	Vector containing the lower bound of the bootstrap confidence interval of the RSEs for the optimal protocol.
RSE_sup_boot_opt_t	Vector containing the upper bound of the bootstrap confidence interval of the RSEs for the optimal protocol.
det_norm_FIM_opt_t	Normalized determinant of the FIM for the optimal protocol.
IC_normal_opt_t	Vector containing the lower and upper bound of the confidence interval of the normalized determinant of the FIM assuming normal distribution for the optimal protocol.
IC_boot_opt_t	Vector containing the lower and upper bound of the bootstrap confidence interval of the normalized determinant of the FIM for the optimal protocol.
list_select	A list containing at each step the combinations of sampling times selected.
list_det	A list containing at each step the determinant of the FIM for each combination of sampling times selected.
list_boot_inf	A list containing at each step the lower bound of the bootstrap confidence interval of determinant of the FIM for each combination of sampling times selected.
list_boot_sup	A list containing at each step the upper bound of the bootstrap confidence interval of determinant of the FIM for each combination of sampling times selected.
list_fim	A list containing at each step the FIM for each combination of sampling times selected.

In addition, for the optimal protocol selected at the end, a graph of the normalized determinant of the FIM with normal and bootstrap confidence intervals in function of the number of MC samples is plotted.

Author(s)

Marie-Karelle Riviere-Jourdan <eldamjh@gmail.com>

Examples

```
#####
# PLEASE UNCOMMENT EXAMPLE #
#####
#params = c(1,8,0.15,0.6,0.02,0.07,0.1)
#times_poss = c(0.1,0.5,1,2,3,6,12,24,36,48,72,120,240)

# Files can be found in external data
#model = stan_model("model_b_given_y.stan")
#model2 = stan_model("model_derivatives.stan")
#model3 = stan_model("model_y.stan")

#opt1 = fisher_optimization(nb_t=3, set_t=times_poss, y_ini=0.5, model=model,
#model2=model2, model3=model3, params=params, dim_b=3, set_seed=TRUE, seed=42,
#step_mc=100, n_samp_min=30, n_samp_max=2000, n_rep=1, n_iter=500, n_burn=500,
#nb_patients=32)
#opt1
```

template_model

Creation of a Pre-Filled Template for STAN Model.

Description

template_model is used to create a pre-filled template for the STAN models used in fisher_evaluation.

Usage

```
template_model(path=getwd(), dloglik, nb_t, outcome, nb_params, ind_RE,
Cov_list=list(), Sigma_b=FALSE, n_rep=1, name)
```

Arguments

path	[Optional] String containing the path where the text file with the model should be created.
dloglik	Boolean indicating if the model is used to calculate the partial derivatives of the log-likelihood (TRUE), or if the model is used to sample in the conditional distribution of b given y (FALSE).
nb_t	Number of sampling times (or doses).
outcome	String indicating the type of outcome. For now, "continuous", "binary", "longitudinal_binary", "count" and "time_to_event" are available.
nb_params	Number of parameters.
ind_RE	Indices for the variance of the random effects in the vector of parameters. It is assumed that the vector of parameters is filled as follows: fixed effects, variances of the random effects, standard deviations of the residual errors (if outcome is "continuous").

Cov_list	[Optional] A list of vectors to specify covariances between random effects. Each element of the list must contain a vector with: the row position, the column position and the value of the covariance in the variance-covariance matrix for the random effects.
Sigma_b	Boolean indicating if the residual errors matrix depends on the random effects b. The default value is set at FALSE.
n_rep	Integer representing the number of repeated measures at the same time (or dose) for each patient. The default value is set at 1 (for "continuous" outcome).
name	String to name the output text file.

Value

Create a text file containing a pre-filled template for the STAN models used in `fisher_evaluation` in the chosen directory.

Author(s)

Marie-Karelle Riviere-Jourdan <eldamjh@gmail.com>

References

Riviere, M-K., Ueckert, S. and Mentre, F. Evaluation of the Fisher information matrix in nonlinear mixed effect models using Markov Chains Monte Carlo.

Examples

```
# UNCOMMENT EXAMPLES
#test2 = template_model(dloglik=TRUE, nb_t=13, outcome="binary",
#nb_params=3, ind_RE=c(3), n_rep=1, name="test2")

#test3 = template_model(dloglik=TRUE, nb_t=8, outcome="continuous",
#nb_params=13, ind_RE=c(5,6,7,8), Cov_list =
#list(c(1,2,0.06),c(2,1,0.06),c(1,3,0.04)), Sigma_fun_b=FALSE, name="test3")

#test4 = template_model( dloglik=FALSE, nb_t=4, outcome="count",
#nb_params=4, ind_RE=c(3,4), n_rep=30, name="test4")

#test5 = template_model(dloglik=FALSE, nb_t=0,
#outcome="time_to_event", nb_params=2, ind_RE=c(2), n_rep=10, name="test5")

#test6 = template_model(dloglik=FALSE, nb_t=4, outcome="continuous",
#nb_params=4, ind_RE=c(3), Sigma_fun_b=FALSE, name="test6")

#test = template_model(dloglik=FALSE, nb_t=7, outcome="continuous",
#nb_params=7, ind_RE=c(4,5,6), Sigma_fun_b=TRUE, n_rep=1, name="test_pk")
```

Index

`fisher_evaluation`, [2](#)

`fisher_optimization`, [5](#)

`MIXFIM (MIXFIM-package)`, [2](#)

`MIXFIM-package`, [2](#)

`template_model`, [8](#)