

# Package ‘DGP4LCF’

March 8, 2025

**Type** Package

**Title** Dependent Gaussian Processes for Longitudinal Correlated Factors

**Version** 1.0.0.1

**Maintainer** Jiachen Cai <jiachen.cai@mrc-bsu.cam.ac.uk>

**Description** Functionalities for analyzing high-dimensional and longitudinal biomarker data to facilitate precision medicine, using a joint model of Bayesian sparse factor analysis and dependent Gaussian processes. This paper illustrates the method in detail: J Cai, RJB Goudie, C Starr, BDM Tom (2023) <[doi:10.48550/arXiv.2307.02781](https://doi.org/10.48550/arXiv.2307.02781)>.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.2.3

**Imports** GPFDA, Rcpp, factor.switching, mvtnorm, combinat, coda, corrplot, pheatmap, stats

**LinkingTo** Rcpp, RcppArmadillo

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Depends** R (>= 2.10)

**VignetteBuilder** knitr

**NeedsCompilation** yes

**Author** Jiachen Cai [aut, cre]

**Repository** CRAN

**Date/Publication** 2025-03-08 08:58:39 UTC

## Contents

factor_loading_heatmap . . . . .	2
factor_score_trajectory . . . . .	3
gibbs_after_mcem_algorithm . . . . .	4
gibbs_after_mcem_combine_chains . . . . .	5

gibbs_after_mcem_diff_initials . . . . .	6
gibbs_after_mcem_load_chains . . . . .	7
mcm_algorithm . . . . .	7
mcm_cov_plot . . . . .	9
mcm_parameter_setup . . . . .	9
numerics_summary_do_not_need_alignment . . . . .	12
numerics_summary_need_alignment . . . . .	13
sim_fcs_init . . . . .	14
sim_fcs_results_irregular_6_8 . . . . .	14
sim_fcs_results_regular_8 . . . . .	14
sim_fcs_truth . . . . .	15
subject_specific_objects . . . . .	15
table_generator . . . . .	16

<b>Index</b>	<b>17</b>
--------------	-----------

---

factor\_loading\_heatmap

*Displaying significant factor loadings in the heatmap.*

---

## Description

This function is used to visualize results of estimates of factor loadings (in heatmaps).

## Usage

```
factor_loading_heatmap(factor_loading_matrix, heatmap_title)
```

## Arguments

`factor_loading_matrix`

A matrix of dimension (p, k), which stores results for factor loadings.

`heatmap_title` A character. Title for the heatmap.

## Value

A heatmap presenting posterior median estimates of factor loadings.

## Examples

```
# See examples in vignette
vignette("bsfadgp_regular_data_example", package = "DGP4LCF")
```

---

`factor_score_trajectory`*Plotting figures for factor score trajectory.*

---

## Description

This function is used to visualize results of factor score trajectories.

## Usage

```
factor_score_trajectory(  
  factor_score_matrix,  
  factor_index,  
  person_index,  
  trajectory_title,  
  cex_main = 1  
)
```

## Arguments

<code>factor_score_matrix</code>	A matrix of dimension (q, k, n), used to store results for factor scores.
<code>factor_index</code>	A numeric scalar. Index of the factor of interest.
<code>person_index</code>	A numeric scalar. Index of the person of interest.
<code>trajectory_title</code>	A character. Title for the factor trajectory plot.
<code>cex_main</code>	A numeric scalar. Text size of the title.

## Value

Trajectory of the designated person-factor.

## Examples

```
# See examples in vignette  
vignette("bsfadgp_regular_data_example", package = "DGP4LCF")
```

---

`gibbs_after_mcem_algorithm`

*Generating posterior samples for parameters (other than DGP parameters) in the model and predicted gene expression for one chain.*

---

### Description

Generating posterior samples for parameters (other than DGP parameters) in the model and predicted gene expression for one chain.

### Usage

```
gibbs_after_mcem_algorithm(
  chain_index,
  mc_num,
  burnin,
  thin_step,
  pathname,
  pred_indicator = FALSE,
  pred_time_index = NULL,
  x,
  mcem_parameter_setup_result,
  mcem_algorithm_result,
  gibbs_after_mcem_diff_initials_result
)
```

### Arguments

<code>chain_index</code>	A numeric scalar. Index of the chain.
<code>mc_num</code>	A numeric scalar. Number of iterations in the Gibbs sampler.
<code>burnin</code>	A numeric scalar. Number of iterations to be discarded as 'burn-in'.
<code>thin_step</code>	A numeric scalar. This function will only save every 'thin_step'th iteration results in the specified directory to reduce storage space needed. Note that this number can be different from that used in the function 'mcem_algorithm'.
<code>pathname</code>	A character. The directory where the saved Gibbs samplers are stored.
<code>pred_indicator</code>	A logical value. <code>pred_indicator = TRUE</code> denotes the need to predict gene expression at new time points. The default value is <code>FALSE</code> .
<code>pred_time_index</code>	Only needed if <code>pred_indicator = TRUE</code> . Index of the new time points in the full time vector.
<code>x</code>	A list of <code>n</code> elements. Each element is a matrix of dimension $(p, q_i)$ , storing the gene expression observed at $q_i$ time points for the $i$ th subject.
<code>mcem_parameter_setup_result</code>	A list of objects returned from the function 'mcem_parameter_setup'.

mcem\_algorithm\_result

A list of objects returned from the function 'mcem\_algorithm'.

gibbs\_after\_mcem\_diff\_initials\_result

A list of objects returned from the function 'gibbs\_after\_mcem\_diff\_initials'.

### Details

This function corresponds to Algorithm 2: Step 1 in the main manuscript; therefore reader can consult the paper for more explanations.

### Value

Posterior samples for parameters (other than DGP parameters) in the model and predicted gene expression for one chain.

### Examples

```
# See examples in vignette
vignette("bsfadgp_regular_data_example", package = "DGP4LCF")
vignette("bsfadgp_irregular_data_example", package = "DGP4LCF")
```

---

gibbs\_after\_mcem\_combine\_chains

*Combining from all chains the posterior samples for parameters in the model and predicted gene expressions.*

---

### Description

Combining from all chains the posterior samples for parameters in the model and predicted gene expressions.

### Usage

```
gibbs_after_mcem_combine_chains(tot_chain, gibbs_after_mcem_algorithm_result)
```

### Arguments

tot\_chain A numeric scalar. Total number of chains.

gibbs\_after\_mcem\_algorithm\_result

A list of objects storing model constants. Should be the same as that input to the 'function gibbs\_after\_mcem\_load\_chains'.

### Value

All saved posterior samples for parameters in the model and predicted gene expressions.

**Examples**

```
# See examples in vignette
vignette("bsfadgp_regular_data_example", package = "DGP4LCF")
```

---

```
gibbs_after_mcem_diff_initials
  Generating different initials for multiple chains.
```

---

**Description**

Generating different initials for multiple chains.

**Usage**

```
gibbs_after_mcem_diff_initials(
  ind_x = TRUE,
  tot_chain = 5,
  mcem_parameter_setup_result,
  mcem_algorithm_result
)
```

**Arguments**

**ind\_x** A logical value. `ind_x = TRUE` uses the model including the intercept term for subject-gene mean in after-MCEM-Gibbs sampler; otherwise uses the model without the intercept term.

**tot\_chain** A numeric scalar. Number of parallel chains.

**mcem\_parameter\_setup\_result**  
A list of objects returned from the function 'mcem\_parameter\_setup'.

**mcem\_algorithm\_result**  
A list of objects returned from the function 'mcem\_algorithm'.

**Value**

Different initials for multiple chains.

**Examples**

```
# See examples in vignette
vignette("bsfadgp_regular_data_example", package = "DGP4LCF")
vignette("bsfadgp_irregular_data_example", package = "DGP4LCF")
```

---

 gibbs\_after\_mcem\_load\_chains

*Loading the saved posterior samples for parameters in the model and predicted gene expressions.*

---

### Description

Loading the saved posterior samples for parameters in the model and predicted gene expressions.

### Usage

```
gibbs_after_mcem_load_chains(chain_index, gibbs_after_mcem_algorithm_result)
```

### Arguments

chain\_index      A numeric scalar. Index of the chain.  
 gibbs\_after\_mcem\_algorithm\_result  
                   A list of objects storing model constants.

### Value

All saved posterior samples for parameters in the model and predicted gene expressions.

### Examples

```
# See examples in vignette
vignette("bsfadgp_regular_data_example", package = "DGP4LCF")
```

---

 mcem\_algorithm

*Monte Carlo Expectation Maximization (MCEM) algorithm to return the Maximum Likelihood Estimate (MLE) of DGP Parameters.*

---

### Description

This function is used to return the MLE of DGP parameters.

### Usage

```
mcem_algorithm(  
  ind_x,  
  ig_parameter = 10^-2,  
  increasing_rate = 0.5,  
  probb_conf_interval = 0.9,  
  iter_count_num = 5,  
  x,
```

```

    mcem_parameter_setup_result,
    ipt_x = FALSE,
    missing_list = NULL,
    missing_num = NULL
  )

```

### Arguments

**ind\_x** A logical value. `ind_x = TRUE` uses the model including the intercept term for subject-gene mean in within-MCEM-Gibbs sampler; otherwise uses the model without the intercept term.

**ig\_parameter** A numeric scalar. Hyper-parameters for the prior Inverse-Gamma distribution.

**increasing\_rate** A numeric scalar. Rate of increasing the sample size.

**prob\_conf\_interval** A numeric scalar. The probability that the true change in the Q-function is larger than the lower bound.

**iter\_count\_num** A numeric scalar. Maximum number of increasing the sample size; a larger number than this would end the algorithm.

**x** A list of  $n$  elements. Each element is a matrix of dimension  $(p, q_i)$ , storing the gene expression observed at  $q_i$  time points for the  $i$ th subject.

**mcem\_parameter\_setup\_result** A list of objects returned from the function `'mcem_parameter_setup'`.

**ipt\_x** A logical value. `ind_x = TRUE` denotes the need to impute for NAs of gene expression. The default value is `ind_x = FALSE`.

**missing\_list** A list of  $n$  elements. Each element is a matrix of dimension  $(\text{missing\_num}, 2)$ : each row corresponds to the position of one NA that needs imputation; first and second columns denote the row and column indexes, respectively, of the NA in the corresponding person's matrix of gene expression.

**missing\_num** A vector of  $n$  elements. Each element corresponds to a single person's number of NAs that needs imputation.

### Value

The MLE of DGP parameters.

### Examples

```

# See examples in vignette
vignette("bsfadgp_regular_data_example", package = "DGP4LCF")
vignette("bsfadgp_irregular_data_example", package = "DGP4LCF")

```



---

mcm_cov_plot	<i>Visualizing cross-correlations among factors.</i>
--------------	--

---

### Description

Visualizing cross-correlations among factors.

### Usage

```
mcm_cov_plot(k, q, cov_input, title)
```

### Arguments

k	A numeric scalar. Number of latent factors.
q	A numeric scalar. Number of time points in the covariance matrix of factors.
cov_input	A matrix of dimension (kq, kq). The covariance matrix of the vector obtained from vectorizing the matrix of latent factor scores.
title	A character. Title for the plot.

### Value

Visualization of cross-correlations among factors.

### Examples

```
# See examples in vignette
vignette("bsfadgp_regular_data_example", package = "DGP4LCF")
vignette("bsfadgp_irregular_data_example", package = "DGP4LCF")
```

---

mcm_parameter_setup	<i>Parameters' setup and initial value assignment for the Monte Carlo Expectation Maximization (MCEM) algorithm.</i>
---------------------	--

---

### Description

This function is used to create R objects storing parameters in the desired format, and assign initial values so that they are ready to use in the MCEM algorithm.

**Usage**

```

mcem_parameter_setup(
  p,
  k,
  n,
  q,
  ind_num = 10,
  burn_in_prop = 0.2,
  thin_step = 5,
  prior_sparsity = 0.1,
  em_num = 50,
  obs_time_num,
  obs_time_index,
  a_person,
  col_person_index,
  y_init,
  a_init,
  z_init,
  phi_init,
  a_full,
  train_index,
  x,
  model_dgp = TRUE
)

```

**Arguments**

p	A numeric scalar. Number of genes.
k	A numeric scalar. Number of latent factors.
n	A numeric scalar. Number of subjects.
q	A numeric scalar. Complete number of time points in the training data.
ind_num	A numeric scalar. Starting size of approximately independent samples for MCEM.
burn_in_prop	A numeric scalar. Proportion of burnin, which be used to calculate size of Monte Carlo samples needed in the Gibbs sampler. Must be the same as that in the function 'mcem_algorithm_irregular_time'.
thin_step	A numeric scalar. Thinning step, which be used to calculate size of Monte Carlo samples needed in the Gibbs sampler. Must be the same as that in the function 'mcem_algorithm_irregular_time'.
prior_sparsity	A numeric scalar. Prior expected proportion of genes involved within each pathway.
em_num	A numeric scalar. Maximum iterations of the expectation maximization (EM) algorithm allowed.
obs_time_num	A n-dimensional vector. One element represents one person's observed number of time points in the training data.
obs_time_index	A list of n elements. One element is a vector of observed time indexes for one person in the training data, sorted from early to late.

a_person	A list of n elements. One element is a vector of observed time for one subject in the training data, sorted from early to late.
col_person_index	A list of n elements. One element is a vector of column indexes for one subject in y_init.
y_init	A matrix of dimension (k, sum(obs_time_num)). Initial values of the latent factor score. Can be obtained using BFRM software.
a_init	A matrix of dimension (p, k). Initial values of the regression coefficients of factor loadings. Can be obtained using BFRM software.
z_init	A matrix of dimension (p, k). Initial values of the binary variables of factor loadings. Can be obtained using BFRM software.
phi_init	A p-dimensional column vector. Initial values of the variance for residuals when modeling gene expressions, corresponding to $\frac{1}{\phi^2}$ in the manuscript. Can be obtained using BFRM software.
a_full	A numeric vector. Complete time observed, sorted from early to late.
train_index	A q-dimensional column vector. Index of time points used in the training data.
x	A list of n elements. Each element is a matrix of dimension (p, q_i), storing the gene expressions for the ith subject.
model_dgp	A logical value. model_dgp = TRUE (default setting) uses the Dependent Gaussian Process to model latent factor trajectories, otherwise the Independent Gaussian Process is used.

### Details

The following parameters are worth particular attention, and users should tune these parameters according to the specific data.

'burn\_in\_prop' and 'thin\_step' co-control the number of Gibbs samples needed in order to generate approximately 'ind\_num' independent samples. The ultimate purpose of tuning these two parameters is to generate high-quality posterior samples for latent factor scores. Therefore: if initials of the Gibbs sampler are not good, readers may need to increase 'burn\_in\_prop' to discard more burn-in samples; if high-correlation is a potential concern, 'thin\_step' may need to be larger.

### Value

A list of R objects required in the MCEM algorithm.

### Examples

```
# See examples in vignette
vignette("bsfadgp_regular_data_example", package = "DGP4LCF")
vignette("bsfadgp_irregular_data_example", package = "DGP4LCF")
```

---

numerics\_summary\_do\_not\_need\_alignment

*Numerical summary for important continuous variables that do not need alignment.*

---

### Description

Numerical summary for important continuous variables that do not need alignment.

### Usage

```
numerics_summary_do_not_need_alignment(
  burnin = 0,
  thin_step = 1,
  pred_x_truth_indicator = FALSE,
  pred_x_truth = NULL,
  gibbs_after_mcem_combine_chains_result
)
```

### Arguments

burnin	A numeric scalar. The saved samples are already after burnin; therefore the default value for this parameter here is 0. Can discard further samples if needed.
thin_step	A numeric scalar. The saved samples are already after thinning; therefore the default value for this parameter here is 1. Can be further thinned if needed.
pred_x_truth_indicator	A logical value. <code>pred_x_truth_indicator = TRUE</code> means that truth of predicted gene expressions are available. The default value is <code>FALSE</code> .
pred_x_truth	Only needed if <code>pred_x_truth_inidicator = TRUE</code> . An array of dimension (n, p, num_time_test), storing true gene expressions in the testing data.
gibbs_after_mcem_combine_chains_result	A list of objects returned from the function <code>'gibbs_after_mcem_combine_chains'</code> .

### Details

This function corresponds to Algorithm 2: Steps 3 and 4 in the main manuscript; therefore reader can consult the paper for more explanations.

### Value

Convergence assessment for important continuous variables that do not need alignment, and posterior summary for predicted gene expressions.

### Examples

```
# See examples in vignette
vignette("bsfadgp_regular_data_example", package = "DGP4LCF")
```

---

`numerics_summary_need_alignment`

*Numerical summary for factor loadings and factor scores, which need alignment.*

---

## Description

Numerical summary for factor loadings and factor scores, which need alignment.

## Usage

```
numerics_summary_need_alignment(  
  burnin = 0,  
  thin_step = 1,  
  gibbs_after_mcem_combine_chains_result  
)
```

## Arguments

`burnin` A numeric scalar. The saved samples are already after burnin; therefore the default value for this parameter here is 0. Can discard further samples if needed.

`thin_step` A numeric scalar. The saved samples are already after thinning; therefore the default value for this parameter here is 1. Can be further thinned if needed.

`gibbs_after_mcem_combine_chains_result`  
A list of objects returned from the function 'gibbs\_after\_mcem\_combine\_chains'.

## Details

This function corresponds to Algorithm 2: Steps 2, 3 and 4 in the main manuscript; therefore reader can consult the paper for more explanations.

## Value

Reordered posterior samples, convergence assessment, and summarized posterior results for factor loadings and factor scores.

## Examples

```
# See examples in vignette  
vignette("bsfadgp_regular_data_example", package = "DGP4LCF")  
vignette("bsfadgp_irregular_data_example", package = "DGP4LCF")
```

---

sim_fcs_init	<i>Initials values.</i>
--------------	-------------------------

---

**Description**

Initial values provided by the two-step approach.

**Usage**

sim\_fcs\_init

**Format**

An object of class list of length 14.

---

sim_fcs_results_irregular_6_8	<i>Results when people have irregularly observed time points (some 6 while others 8).</i>
-------------------------------	---

---

**Description**

Results when people have irregularly observed time points (some 6 while others 8).

**Usage**

sim\_fcs\_results\_irregular\_6\_8

**Format**

An object of class list of length 3.

---

sim_fcs_results_regular_8	<i>Results when people are observed at common 8 time points.</i>
---------------------------	--

---

**Description**

Results when people are observed at common 8 time points.

**Usage**

sim\_fcs\_results\_regular\_8

**Format**

An object of class list of length 3.

---

sim_fcs_truth	<i>Truth of simulated data.</i>
---------------	---------------------------------

---

**Description**

Simulated data under the scenario where factors are correlated and have small variability (CS).

**Usage**

```
sim_fcs_truth
```

**Format**

An object of class list of length 19.

---

subject_specific_objects	<i>Constructing subject-specific objects required for Gibbs sampler (for subjects with incomplete observations only).</i>
--------------------------	---

---

**Description**

Constructing subject-specific objects required for Gibbs sampler (for subjects with incomplete observations only).

**Usage**

```
subject_specific_objects(k, q, a_full, a_avail, cor_all)
```

**Arguments**

k	A numeric scalar. Number of latent factors.
q	A numeric scalar. Number of time points in the complete factor covariance matrix.
a_full	A q-dimensional numeric vector. Complete time sorted from early to late.
a_avail	A vector of time when gene expressions are available, sorted from early to late.
cor_all	A matrix of dimension (kq, kq). Correlation matrix of latent factor scores.

**Details**

This function is used to extract subject-specific factor covariance matrix from the complete factor covariance matrix, through constructing subject-specific indicator matrix, which indicates time indexes when gene expression are available.

**Value**

Subject-specific objects needed for Gibbs sampler.

**Examples**

```
# See examples in vignette
vignette("bsfadgp_regular_data_example", package = "DGP4LCF")
vignette("bsfadgp_irregular_data_example", package = "DGP4LCF")
```

---

table_generator	<i>Generating a table listing all possible combinations of the binary variables for one gene.</i>
-----------------	---

---

**Description**

Generating a table listing all possible combinations of the binary variables for one gene.

**Usage**

```
table_generator(k)
```

**Arguments**

k                    A numeric scalar. Number of latent factors.

**Value**

A table listing all possible combinations of the binary variables for one gene.

**Examples**

```
# See examples in vignette
vignette("bsfadgp_regular_data_example", package = "DGP4LCF")
vignette("bsfadgp_irregular_data_example", package = "DGP4LCF")
```



# Index

## \* datasets

- [sim\\_fcs\\_init](#), [14](#)
- [sim\\_fcs\\_results\\_irregular\\_6\\_8](#), [14](#)
- [sim\\_fcs\\_results\\_regular\\_8](#), [14](#)
- [sim\\_fcs\\_truth](#), [15](#)

- [factor\\_loading\\_heatmap](#), [2](#)
- [factor\\_score\\_trajectory](#), [3](#)

- [gibbs\\_after\\_mcem\\_algorithm](#), [4](#)
- [gibbs\\_after\\_mcem\\_combine\\_chains](#), [5](#)
- [gibbs\\_after\\_mcem\\_diff\\_initials](#), [6](#)
- [gibbs\\_after\\_mcem\\_load\\_chains](#), [7](#)

- [mcm\\_algorithm](#), [7](#)
- [mcm\\_cov\\_plot](#), [9](#)
- [mcm\\_parameter\\_setup](#), [9](#)

- [numerics\\_summary\\_do\\_not\\_need\\_alignment](#),  
[12](#)
- [numerics\\_summary\\_need\\_alignment](#), [13](#)

- [sim\\_fcs\\_init](#), [14](#)
- [sim\\_fcs\\_results\\_irregular\\_6\\_8](#), [14](#)
- [sim\\_fcs\\_results\\_regular\\_8](#), [14](#)
- [sim\\_fcs\\_truth](#), [15](#)
- [subject\\_specific\\_objects](#), [15](#)

- [table\\_generator](#), [16](#)