

Package: ProteoBayes (via r-universe)

October 16, 2026

Title Bayesian Statistical Tools for Quantitative Proteomics

Version 1.0.0

Description Bayesian toolbox for quantitative proteomics. In particular, this package provides functions to generate synthetic datasets, execute Bayesian differential analysis methods and display results, as described in the associated article Marie Chion and Arthur Leroy (2025) <[doi:10.48550/arXiv.2307.08975](https://doi.org/10.48550/arXiv.2307.08975)>.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Imports dplyr, ggplot2, magrittr, mvtnorm, tibble, tidyr, rlang, extraDistr, gridExtra, LaplacesDemon

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

URL <https://mariechion.github.io/ProteoBayes/>

Config/pak/sysreqs libicu-dev

Repository <https://mariechion.r-universe.dev>

Date/Publication 2026-04-19 15:34:08 UTC

RemoteUrl <https://github.com/mariechion/proteobayes>

RemoteRef HEAD

RemoteSha d7a72f71de3d6ea021a374d92dfd355261f14343

Contents

identify_diff	2
multi_CI	3
multi_identify_diff	3
multi_overlap_coef	5
multi_posterior_mean	5

overlap_coef	6
plot_distrib	7
plot_multi_diff	8
posterior_mean	9
sample_distrib	10
simu_db	11
vectorised_multi	12

Index	13
--------------	-----------

identify_diff	<i>Identify posterior mean differences</i>
---------------	--

Description

Compute a criterion based on Credible Intervals (CI) to determine whether the posterior t-distributions of groups should be considered different enough to deserve further examination. Two groups are considered probably 'distinct' if the Credible Interval of level CI_level of their respective posterior t-distributions do not overlap.

Usage

```
identify_diff(posterior)
```

Arguments

posterior	A tibble, typically coming from a posterior_mean() function, containing the parameters of the multivariate posterior t-distributions for the mean of the considered groups and draws for each peptide.
-----------	--

Value

A tibble, indicating which peptides and groups seem to be different

Examples

```
TRUE
```

multi_CI

*Multidimensional Credible Interval***Description**

Compute whether a given vector belongs to the multivariate t-distribution Credible Interval for given level, mean, covariance and degrees of freedom.

Usage

```
multi_CI(data, mean, cov, df, level = 0.95)
```

Arguments

data	A vector, of compatible dimension with mean and cov
mean	A vector, the mean parameter of the multivariate t-distribution
cov	A matrix, the covariance parameter of the multivariate t-distribution
df	A number, the degrees of freedom of the multivariate t-distribution
level	A number, between 0 and 1, corresponding to the level of the Credible Interval. Default is 0.95.

Value

A boolean, indicating whether the data vector belongs to the computed Credible Interval.

Examples

```
TRUE
```

multi_identify_diff

*Identify differences in multivariate posteriors***Description**

Compute a multivariate inference criterion to examine whether the posterior multivariate t-distributions of groups should be considered different enough to be called 'differential'. Two groups are considered can be discriminated based on the probability weights for the element-wise means to be greater in each group. The Credible Intervals for each marginals are also provided.

Usage

```
multi_identify_diff(  
  posterior,  
  plot = TRUE,  
  overlap_coef = TRUE,  
  cumulative = FALSE,  
  nb_sample = 1e+05,  
  nb_sample_overlap = 10000  
)
```

Arguments

posterior	A tibble, typically coming from a <code>posterior_mean()</code> function, containing the parameters of the multivariate posterior t-distributions for the mean of the considered groups and draws for each peptide.
plot	A boolean, indicating whether a results plot should be displayed.
overlap_coef	A boolean, indicating whether the overlapping coefficient between multivariate t-distributions should be computed for all groups.
cumulative	A boolean, indicating whether the probability distribution should be the cumulative instead.
nb_sample	A number of samples to draw from the empirical distributions
nb_sample_overlap	A number of samples to draw for the Monte Carlo approximation of the Overlapping Coefficients between all posteriors.

Value

A list, containing:

- `Diff_mean`, a tibble containing the posterior mean (and their difference) for each peptide and for all one-by-one group comparisons.
- `Diff_proba`, a tibble containing the probability distribution of the number of differential peptides for all one-by-one group comparisons.
- `Overlap_coef`, a tibble containing the overlapping coefficient between the posterior multivariate t-distributions for all one-by-one group comparisons.

Examples

```
TRUE
```

multi_overlap_coef	<i>Overlapping coefficient between multivariate t-distributions</i>
--------------------	---

Description

Compute a Monte Carlo approximation of the overlapping coefficient between two multivariate t-distributions with arbitrary mean, covariance and degrees of freedom.

Usage

```
multi_overlap_coef(mean1, mean2, cov1, cov2, df1, df2, nb_sample = 10000)
```

Arguments

mean1	A vector, the mean parameter of a multi t-distribution
mean2	A vector, the mean parameter of the other multi t-distribution
cov1	A matrix, the covariance parameter of a multi t-distribution
cov2	A matrix, the covariance parameter of the other multi t-distribution
df1	A number, the degrees of freedom of a multi t-distribution
df2	A number, the degrees of freedom of the other multi t-distribution
nb_sample	A number of samples drawn to compute the Monte Carlo estimation

Value

A number, the Monte Carlo approximation of the overlapping coefficient between the two multivariate t-distributions.

Examples

```
TRUE
```

multi_posterior_mean	<i>Multivariate posterior distribution of the means</i>
----------------------	---

Description

Compute the multivariate posterior distribution of the means between multiple groups, for multiple correlated peptides. The function accounts for multiple imputations through the Draw identifier in the dataset.

Usage

```
multi_posterior_mean(
  data,
  mu_0 = NULL,
  lambda_0 = 1,
  Sigma_0 = NULL,
  nu_0 = 10,
  vectorised = FALSE
)
```

Arguments

data	A tibble or data frame containing imputed data sets for all groups. Required columns: Peptide, Group, Sample, Output. If missing data have been estimated from multiple imputations, each imputation should be identified in an optional Draw column.
mu_0	A vector, corresponding to the prior mean. If NULL, all groups are initialised with the same empirical mean for each peptide.
lambda_0	A number, corresponding to the prior covariance scaling parameter.
Sigma_0	A matrix, corresponding to the prior covariance parameter. If NULL, the identity matrix will be used by default.
nu_0	A number, corresponding to the prior degrees of freedom.
vectorised	A boolean, indicating whether we should used a vectorised version of the function. Default when nb_peptides < 30. If nb_peptides > 30, there is a high risk that the vectorised version would be slower.

Value

A tibble providing the parameters of the multivariate posterior t-distribution for the mean of the considered groups and draws for each peptide.

Examples

```
TRUE
```

overlap_coef	<i>Overlapping coefficient between univariate t-distributions</i>
--------------	---

Description

Compute a (high speed) quadrature approximation of the overlapping coefficient between two univariate t-distributions with arbitrary mean, covariance and degrees of freedom.

Usage

```
overlap_coef(mean1, mean2, var1, var2, df1, df2)
```

Arguments

mean1	A vector, the mean parameter of a t-distribution
mean2	A vector, the mean parameter of the other t-distribution
var1	A matrix, the variance parameter of a t-distribution
var2	A matrix, the variance parameter of the other t-distribution
df1	A number, the degrees of freedom of a t-distribution
df2	A number, the degrees of freedom of the other t-distribution
nb_sample	A number of samples drawn to compute the Monte Carlo estimation

Value

A number, the Monte Carlo approximation of the overlapping coefficient between the two univariate t-distributions.

Examples

```
TRUE
```

plot_distrib	<i>Plot the posterior distribution of the difference of means</i>
--------------	---

Description

Display the posterior distribution of the difference of means between two groups for a specific peptide. If only one group is provide, the function display the posterior distribution of the mean for this specific group instead. The function provides additional tools to represent information to help inference regarding the difference between groups (reference at 0 on the x-axis, probability of group1 > group2 and conversely).

Usage

```
plot_distrib(  
  sample_distrib,  
  group1 = NULL,  
  group2 = NULL,  
  peptide = NULL,  
  prob_CI = 0.95,  
  show_prob = TRUE,  
  mean_bar = TRUE,  
  index_group1 = NULL,  
  index_group2 = NULL  
)
```

Arguments

sample_distrib	A data frame, typically coming from the sample_distrib() function, containing the following columns: Peptide, Group and Sample. This argument should contain the empirical posterior distributions to be displayed.
group1	A character string, corresponding to the name of the group for which we plot the posterior distribution of the mean. If NULL (default), the first group appearing in sample_distrib is displayed. If group2 is provided, the posterior difference of the groups is displayed instead.
group2	A character string, corresponding to the name of the group we want to compare to group1. If NULL (default), only the posterior distribution of the mean for group1 is displayed.
peptide	A character string, corresponding to the name of the peptide for which we plot the posterior distribution of the mean. If NULL (default), only the first appearing in sample_distrib is displayed.
prob_CI	A number, between 0 and 1, corresponding the level of the Credible Interval (CI), represented as side regions (in red) of the posterior distribution. The default value (0.95) display the 95% CI, meaning that the central region (in blue) contains 95% of the probability distribution of the mean.
show_prob	A boolean, indicating whether we display the label of probability comparisons between the two groups.
mean_bar	A boolean, indicating whether we display the vertical bar corresponding to 0 on the x-axis (when comparing two groups), of the mean value of the distribution (when displaying a unique group).
index_group1	A character string, used as the index of group1 in the legends. If NULL (default), group1 is used.
index_group2	A character string, used as the index of group2 in the legends. If NULL (default), group2 is used.

Value

Plot of the required posterior distribution.

Examples

```
TRUE
```

plot_multi_diff	<i>Plot multivariate comparison between groups</i>
-----------------	--

Description

Graphical representation of inference in a multivariate difference analysis context. The plotted empirical distribution represents, for each one-to-one group comparison, the probability of the number of elements for which the mean of a peptide is larger in a given group. This provides visual evidence on whether two groups are differential or not, with adequate uncertainty quantification.

Usage

```
plot_multi_diff(multi_diff, plot_mean = TRUE, cumulative = FALSE)
```

Arguments

multi_diff	A tibble, typically coming from the multi_identify_diff function, containing probability distribution (or the cumulative distribution) of the number of larger Peptides in each one-to-one group comparisons.
plot_mean	A boolean, indicating whether the graph for difference of means between all groups for each Peptide should be displayed.
cumulative	A boolean, indicating whether the cumulative distribution or the original probability distribution should be displayed.

Value

A graph (or a matrix of graphs) displaying the multivariate differential inference summary between groups

Examples

```
TRUE
```

posterior_mean	<i>Posterior distribution of the means</i>
----------------	--

Description

Compute the posterior distribution of the means between multiple groups. All peptides are considered independent from one another.

Usage

```
posterior_mean(data, mu_0 = NULL, lambda_0 = 1, beta_0 = 1, alpha_0 = 1)
```

Arguments

data	A tibble or data frame containing imputed data sets for all groups. Required columns: Peptide, Output, Group, Sample.
mu_0	A vector, corresponding to the prior mean.
lambda_0	A number, corresponding to the prior covariance scaling parameter.
beta_0	A matrix, corresponding to the prior covariance parameter.
alpha_0	A number, corresponding to the prior degrees of freedom.

Value

A tibble providing the empirical posterior distribution for the

Examples

```
TRUE
```

sample_distrib	<i>Sample from a t-distribution</i>
----------------	-------------------------------------

Description

Sample from a (possibly multivariate) t-distribution. This function can be used to sample both from a prior or posterior, depending on the value of parameters provided.

Usage

```
sample_distrib(posterior, nb_sample = 1000)
```

Arguments

posterior	A tibble or data frame, detailing for each Peptide and each Group, the value of the t-distribution parameters. The expected format is typically a return from a posterior_mean() function. Expected columns in the univariate case: mu, lambda, alpha, beta. Expected columns in the multivariate case: mu, lambda, Sigma, nu.
nb_sample	A number, indicating the number of samples generated for each couple Peptide-Group.

Value

A tibble containing the Peptide, Group and Sample columns. The samples of each Peptide-Group couple provide an empirical t-distribution that can be used to compute and display differences between groups.

Examples

```
TRUE
```

simu_db

*Generate a synthetic dataset tailored for ProteoBayes***Description**

Simulate a complete training dataset, which may be representative of various applications. Several flexible arguments allow adjustment of the number of peptides, of groups, and samples in each experiment. The values of several parameters controlling the data generation process can be modified.

Usage

```
simu_db(
  nb_peptide = 5,
  nb_group = 2,
  nb_sample = 5,
  multi_imp = FALSE,
  nb_draw = 5,
  range_peptide = c(0, 50),
  diff_group = 3,
  var_sample = 2,
  var_draw = 1
)
```

Arguments

nb_peptide	An integer, indicating the number of peptides in the data.
nb_group	An integer, indicating the number of groups/conditions.
nb_sample	An integer, indicating the number of samples in the data for each peptide (i.e the repetitions of the same experiment).
multi_imp	A boolean, indicating whether multiple imputations have been applied to obtain the dataset.
nb_draw	A number, indicating the number of imputation procedures applied to obtain this dataset.
range_peptide	A 2-sized vector, indicating the range of values from which to pick a mean value for each peptide.
diff_group	A number, indicating the mean difference between consecutive groups
var_sample	A number, indicating the noise variance for each new sample of a peptide.
var_draw	A number, indicating the noise variance for each imputation draw.

Value

A full dataset of synthetic data.

Examples

```
## Generate a dataset with 5 peptides in each of the 2 groups, observed for
## 3 different samples
data = simu_db(nb_peptide = 5, nb_group = 2, nb_sample = 3)

## Generate a dataset with 3 peptides in each of the 3 groups, observed for
## 4 different samples, for which 5 imputation draws are available.
data = simu_db(nb_peptide = 3, nb_group = 3, nb_sample = 4, nb_draw = 5)
```

vectorised_multi	<i>Vectorised version of multi_posterior_mean()</i>
------------------	---

Description

Alternative vectorised version, highly efficient when nb_peptide < 30.

Usage

```
vectorised_multi(data, mu_0 = NULL, lambda_0 = 1, Sigma_0 = NULL, nu_0 = 10)
```

Arguments

data	A tibble or data frame containing imputed data sets for all groups. Required columns: Peptide, Group, Sample, Output. If missing data have been estimated from multiple imputations, each imputation should be identified in an optional Draw column.
mu_0	A vector, corresponding to the prior mean. If NULL, all groups are initialised with the same empirical mean for each peptide.
lambda_0	A number, corresponding to the prior covariance scaling parameter.
Sigma_0	A matrix, corresponding to the prior covariance parameter. If NULL, the identity matrix will be used by default.
nu_0	A number, corresponding to the prior degrees of freedom.

Value

A tibble providing the parameters of the posterior t-distribution for the mean of the considered groups for each peptide.

Examples

```
TRUE
```

Index

identify_diff, [2](#)

multi_CI, [3](#)

multi_identify_diff, [3](#)

multi_overlap_coef, [5](#)

multi_posterior_mean, [5](#)

overlap_coef, [6](#)

plot_distrib, [7](#)

plot_multi_diff, [8](#)

posterior_mean, [9](#)

sample_distrib, [10](#)

simu_db, [11](#)

vectorised_multi, [12](#)