Package 'pvEBayes'

July 10, 2025

Title Empirical Bayes Methods for Pharmacovigilance

Version 0.1.1

Maintainer Yihao Tan <yihaotan@buffalo.edu>

Description A suite of empirical Bayes methods to use in pharmacovigilance. Contains various model fitting and post-processing functions. For more details see Tan et al. (2025) <doi:10.48550/arXiv.2502.09816>, Koenker and Mizera (2014) <doi:10.1080/01621459.2013.869224> and Efron (2016) <doi:10.1093/biomet/asv068>.

License GPL-3

Encoding UTF-8

RoxygenNote 7.3.2

Depends R (>= 3.6.0)

Imports data.table, ggdist, ggfittext, ggplot2, glue, graphics, magrittr, methods, SobolSequence, stats, wacolors, Rcpp, REBayes, splines

LinkingTo Rcpp, RcppEigen

LazyData true

Suggests testthat (>= 3.0.0), Rmosek

Config/testthat/edition 3

URL https://github.com/YihaoTancn/pvEBayes

BugReports https://github.com/YihaoTancn/pvEBayes/issues

NeedsCompilation yes

Author Yihao Tan [aut, cre] (ORCID: <https://orcid.org/0009-0002-9196-8166>), Marianthi Markatou [aut] (ORCID: <https://orcid.org/0000-0002-1453-8229>), Saptarshi Chakraborty [aut] (ORCID: <https://orcid.org/0000-0002-3121-9174>), Raktim Mukhopadhyay [aut] (ORCID: <https://orcid.org/0009-0007-8972-6682>)

Repository CRAN

Date/Publication 2025-07-10 15:20:06 UTC

Contents

`AIC.pvEBayes` *Obtain Akaike Information Criterion (AIC) for a pvEBayes object*

Description

This function defines the S3 `AIC` method for objects of class `pvEBayes`. It extracts the Akaike Information Criterion (AIC) from a fitted model.

Usage

```
## S3 method for class 'pvEBayes'
AIC(object, ..., k = 2)
```

Arguments

`object`	a `pvEBayes` object, which is the output of the function pvEBayes or pvEBayes_tune.
`...`	other input parameters. Currently unused.
`k`	numeric, the penalty per parameter to be used; the default k = 2 is the classical AIC.

Value

numeric, AIC score for the resulting model.

Examples

```
fit <- pvEBayes(
  contin_table = statin2025_44, model = "general-gamma",
  alpha = 0.3, n_posterior_draws = NULL
)

AIC_score <- AIC(fit)
```

`BIC.pvEBayes` *Obtain Bayesian Information Criterion (BIC) for a pvEBayes object*

Description

This function defines the S3 `BIC` method for objects of class `pvEBayes`. It extracts the Bayesian Information Criterion (BIC) from a fitted model.

Usage

```
## S3 method for class 'pvEBayes'
BIC(object, ...)
```

Arguments

`object`	a `pvEBayes` object, which is the output of the function pvEBayes or pvEBayes_tune.
`...`	other input parameters. Currently unused.

Value

numeric, BIC score for the resulting model.

Examples

```
fit <- pvEBayes(
  contin_table = statin2025_44, model = "general-gamma",
  alpha = 0.3, n_posterior_draws = NULL
)

BIC_score <- BIC(fit)
```

`estimate_null_expected_count` *Estimate expected null baseline count based on reference row and column*

Description

This function estimates the expected null baseline count (E_{ij}) for each AE-drug combination under the assumption of independence between rows and columns. The expected count is calculated using a reference row (other AEs) and reference column (other drugs). This null baseline is typically used in empirical Bayes modeling of `pvEBayes` package for signal detection and estimation in spontaneous reporting system (SRS) data.

Usage

```
estimate_null_expected_count(contin_table)
```

Arguments

`contin_table`	an IxJ contingency table showing pairwise counts of adverse events for I AEs (along the rows) and J drugs (along the columns). The reference row "Other AEs" and the reference column "Other drugs" need to be the I-th row and J-th column respectively.

Details

This null value estimator is proposed by Tan et al. (2025).

Value

an `nrow(contin_table)` by `ncol(contin_table)` matrix.

References

Tan Y, Markatou M and Chakraborty S. Flexible Empirical Bayesian Approaches to Pharmacovigilance for Simultaneous Signal Detection and Signal Strength Estimation in Spontaneous Reporting Systems Data. *arXiv preprint.* 2025; arXiv:2502.09816.

Examples

```
estimate_null_expected_count(statin2025_44)
```

extract_all_fitted_models *Extract all fitted models from a tuned pvEBayes Object*

Description

This function retrieves the list of all fitted models from a pvEBayes_tuned object, which is the output of the `pvEBayes_tune()` function.

Usage

```
extract_all_fitted_models(object)
```

Arguments

`object`	An object of class `pvEBayes_tuned`, usually returned by `pvEBayes_tune`. This function will throw an error if the input is not of the correct class.

Value

A list containing the results of each model fitted during the tuning process.

Examples

```
valid_matrix <- matrix(c(1, 2, 3, 4, 5, 6, 7, 8), nrow = 2)
rownames(valid_matrix) <- c("AE_1", "AE_2")
colnames(valid_matrix) <- c("drug_1", "drug_2", "drug_3", "drug_4")

tuned_object = pvEBayes_tune(valid_matrix,
                             model = "general-gamma",
                             return_all_fit = TRUE)
extract_all_fitted_models(tuned_object)
```

eyeplot_pvEBayes *Generate an eyeplot showing the distribution of posterior draws for selected drugs and adverse events*

Description

This function creates an eyeplot to visualize the posterior distributions of λ_{ij} for selected AEs and drugs. The plot displays posterior median, 90 percent credible interval for each selected AE-drug combination.

Usage

```
eyeplot_pvEBayes(
  x,
  num_top_AEs = 10,
  num_top_drugs = 8,
  specified_AEs = NULL,
  specified_drugs = NULL,
  N_threshold = 1,
  text_shift = 4,
  x_lim_scalar = 1.3,
  text_size = 3,
  log_scale = FALSE
)
```

Arguments

x	a pvEBayes object, which is the output of the function pvEBayes or pvEBayes_tune.
num_top_AEs	number of most significant AEs appearing in the plot. Default to 10.
num_top_drugs	number of most significant drugs appearing in the plot. Default to 7.
specified_AEs	a vector of AE names that are specified to appear in the plot. If a vector of AEs is given, argument num_top_AEs will be ignored.
specified_drugs	a vector of drug names that are specified to appear in the plot. If a vector of drugs is given, argument num_top_drugs will be ignored.
N_threshold	integer greater than 0. Any AE-drug combination with observation smaller than N_threshold will be filtered out.
text_shift	numeric. Controls the relative position of text labels, (e.g., "N = 1", "E = 2"). A larger value shifts the "E = 2" further away from its original position.
x_lim_scalar	numeric. An x-axis range scalar that ensures text labels are appropriately included in the plot.
text_size	numeric. Controls the size of text labels, (e.g., "N = 1", "E = 2").
log_scale	logical. If TRUE, the eye plot displays the posterior distribution of $\log(\lambda_{ij})$ for the selected AEs and drugs.

Value

a ggplot2 object.

Examples

```
fit <- pvEBayes(
  contin_table = statin2025_44, model = "general-gamma",
  alpha = 0.3, n_posterior_draws = 1000
)

AE_names <- rownames(statin2025_44)[1:6]
```

```
drug_names <- colnames(statin2025_44)[-7]

eyeplot_pvEBayes(
  x = fit
)
```

`gbca2025` *FDA GBCA dataset with 1328 adverse events*

Description

An adverse event-drug count dataset (contingency table) obtained from the FDA FAERS database for the quarters 2021Q1 - 2024Q4.

Usage

```
gbca2025
```

Format

An object of class `matrix` (inherits from `array`) with 1328 rows and 8 columns.

Details

Data are stored in the form of a contingency table, with drugs listed across the columns and the 1328 AEs presented across the rows. Each cell in the contingency table represents the total report counts associated with that (AE, drug) pair and detected in the FDA FAERS database during 2021Q1 - 2024Q4.

The dataset catalogs 7 Gadolinium-Based Contrast Agents (GBCAs) (across columns):

Gadobenate, Gadobutrol, Gadodiamide, Gadopentetate, Gadoterate, Gadoteridol, Gadoxetate

The 1328 adverse events presented across the rows are AEs that contain at least one report for the 7 GBCA drugs during 2021Q1 - 2024Q4.

This dataset is an updated version of gbca from the pvLRT package which collects the same scope of AEs for 7 gbca drugs for quarters 2014Q3 - 2020Q4.

Source

https://fis.fda.gov/extensions/FPD-QDE-FAERS/FPD-QDE-FAERS.html

gbca2025_69 *FDA GBCA dataset with 69 adverse events*

Description

An adverse event-drug count dataset (contingency table) obtained from the FDA FAERS database for the quarters 2021Q1 - 2024Q4

Usage

```
gbca2025_69
```

Format

An object of class `matrix` (inherits from `array`) with 70 rows and 8 columns.

Details

Data are stored in the form of a contingency table, with drugs listed across the columns and the 69 AEs presented across the rows. Each cell in the contingency table represents the total report counts associated with that (AE, drug) pair and detected in the FDA FAERS database during 2021Q1 - 2024Q4.

The dataset catalogs 7 Gadolinium-Based Contrast Agents (GBCAs) (across columns):

Gadobenate, Gadobutrol, Gadodiamide, Gadopentetate, Gadoterate, Gadoteridol, Gadoxetate.

The 69 adverse events presented across the rows are selected from 1328 AEs of gbca2025 which are related to the brain or neural system. Other AEs are collapsed to one reference row: "Other AEs".

Source

https://fis.fda.gov/extensions/FPD-QDE-FAERS/FPD-QDE-FAERS.html

generate_contin_table *Generate random contingency tables based on a reference table embedded signals,and possibly with zero inflation*

Description

This function generates random contingency tables that resemble a given reference table, with the option to embed signals and zero-inflation.

Usage

```
generate_contin_table(
  n_table = 1,
  ref_table,
  signal_mat = NULL,
  Variation = FALSE,
  zi_indic_mat = NULL
)
```

Arguments

n_table	number of random matrices to generate.
ref_table	a reference table used as the basis for generating random tables.
signal_mat	numeric matrix of the same dimension as the reference table (ref_table). The entry at position (i, j) in signal_mat represents the signal strength between the i-th adverse event and the j-th drug. By default, each pair is assigned a value of 1, indicating no signal for that pair.
Variation	logical. Include random noises to sig_mat while generating random tables. Default to FALSE. If set to TRUE, n_table of sig_mat incorporating the added noise will also be returned.
zi_indic_mat	logical matrix of the same size as ref_table indicating the positions of structural zero.

Value

A list of length `n_table`, with each entry being a `nrow(ref_table)` by `ncol(ref_table)` matrix.

References

Tan Y, Markatou M and Chakraborty S. Flexible Empirical Bayesian Approaches to Pharmacovigilance for Simultaneous Signal Detection and Signal Strength Estimation in Spontaneous Reporting Systems Data. *arXiv preprint.* 2025; arXiv:2502.09816.

Examples

```
set.seed(1)
ref_table <- statin2025_44


# set up signal matrix with one signal at entry (1,1)
sig_mat <- matrix(1, nrow(ref_table), ncol(ref_table))
sig_mat[1, 1] <- 2

# set up structural zero matrix
Z <- matrix(0, nrow(ref_table), ncol(ref_table))
Z[5, 1] <- 1

simu_table <- generate_contin_table(ref_table,
  signal_mat = sig_mat,
```

```
  n_table = 1,
  Variation = TRUE,
  zi_indic_mat = Z
)[[1]][[1]]
```

`heatmap_pvEBayes` *Generate a heatmap plot visualizing posterior probabilities for selected drugs and adverse events*

Description

This function generates a heatmap to visualize the posterior probabilities of being a signal for selected AEs and drugs.

Usage

```
heatmap_pvEBayes(
  x,
  num_top_AEs = 10,
  num_top_drugs = 8,
  specified_AEs = NULL,
  specified_drugs = NULL,
  cutoff_signal = NULL
)
```

Arguments

`x`	a `pvEBayes` object, which is the output of the function pvEBayes or pvEBayes_tune.
`num_top_AEs`	number of most significant AEs appearing in the plot. Default to 10.
`num_top_drugs`	number of most significant drugs appearing in the plot. Default to 7.
`specified_AEs`	a vector of AE names that are specified to appear in the plot. If a vector of AEs is given, argument num_top_AEs will be ignored.
`specified_drugs`	a vector of drug names that are specified to appear in the plot. If a vector of drugs is given, argument num_top_drugs will be ignored.
`cutoff_signal`	numeric. Threshold for signal detection. An AE-drug combination is classified as a detected signal if its 5th posterior percentile exceeds this threshold.

Value

a ggplot2 object.

Examples

```
fit <- pvEBayes(
  contin_table = statin2025_44, model = "general-gamma",
  alpha = 0.3, n_posterior_draws = 1000
)


heatmap_pvEBayes(
  x = fit,
  cutoff = 1.001
)
```

`logLik.pvEBayes` *Extract log marginal likelihood for a pvEBayes object*

Description

This function defines the S3 `logLik` method for objects of class `pvEBayes`. It extracts the log marginal likelihood from a fitted model.

Usage

```
## S3 method for class 'pvEBayes'
logLik(object, ...)
```

Arguments

`object`	a `pvEBayes` object, which is the output of the function pvEBayes or pvEBayes_tune.
`...`	other input parameters. Currently unused.

Value

returns log marginal likelihood of a pvEBayes object.

Examples

```
fit <- pvEBayes(
  contin_table = statin2025_44, model = "general-gamma",
  alpha = 0.3, n_posterior_draws = NULL
)

logLik(fit)
```

plot.pvEBayes *Plotting method for a pvEBayes object*

Description

This function defines the S3 plot method for objects of class pvEBayes.

Usage

```
## S3 method for class 'pvEBayes'
plot(x, type = "eyeplot", ...)
```

Arguments

x	a pvEBayes object, which is the output of the function pvEBayes or pvEBayes_tune.
type	character string determining the type of plot to show. Available choices are "eyeplot" which calls eyeplot_pvEBayes and "heatmap" which calls heatmap_pvEBayes, with the additional arguments supplied in ...
...	additional arguments passed to heatmap_pvEBayes or eyeplot_pvEBayes.

Value

A ggplot object.

Examples

```
obj <- pvEBayes(statin2025_44, model = "general-gamma", alpha = 0.5)
plot(obj, type = "eyeplot")
```

posterior_draws *Generate posterior draws for each AE-drug combination*

Description

This function generates posterior draws from the posterior distribution of λ_{ij} for each AE-drug combination, based on a fitted empirical Bayes model. The posterior draws can be used to compute credible intervals, visualize posterior distributions, or support downstream inference.

Usage

```
posterior_draws(obj, n_posterior_draws = 1000)
```

Arguments

obj	a pvEBayes object, which is the output of the function pvEBayes or pvEBayes_tune.
n_posterior_draws	number of posterior draws for each AE-drug combination.

Value

The function returns an S3 object of class `pvEBayes` with posterior draws.

Examples

```
fit <- pvEBayes(
  contin_table = statin2025_44, model = "general-gamma",
  alpha = 0.3, n_posterior_draws = NULL
)

fit_with_draws <- posterior_draws(fit, n_posterior_draws = 1000)
```

`print.pvEBayes` *Print method for a pvEBayes object*

Description

This function defines the S3 `print` method for objects of class `pvEBayes`. It displays a concise description of the fitted model.

Usage

```
## S3 method for class 'pvEBayes'
print(x, ...)
```

Arguments

`x`	a `pvEBayes` object, which is the output of the function pvEBayes or pvEBayes_tune.
`...`	other input parameters. Currently unused.

Value

Invisibly returns the input `pvEBayes` object.

Examples

```
obj <- pvEBayes(
  contin_table = statin2025_44, model = "general-gamma",
  alpha = 0.5, n_posterior_draws = 10000
)

print(obj)
```

pvEBayes *Fit a general-gamma, GPS, K-gamma, KM or efron model for a contingency table.*

Description

This function fits a non-parametric empirical Bayes model to an AE-drug contingency table using one of several empirical Bayes approaches with specified hyperparameter, if is required. Supported models include the "general-gamma", "GPS", "K-gamma", "KM", and "efron".

Usage

```
pvEBayes(
  contin_table,
  model = "general-gamma",
  alpha = NULL,
  K = NULL,
  p = NULL,
  c0 = NULL,
  maxi = NULL,
  eps = 1e-04,
  n_posterior_draws = 1000
)
```

Arguments

contin_table	an IxJ contingency table showing pairwise counts of adverse events for I AEs (along the rows) and J drugs (along the columns).
model	the model to fit. Available models are "general-gamma", "K-gamma", "GPS", "KM" and "efron". Default to "general-gamma".
alpha	numeric between 0 and 1. The hyperparameter of "general-gamma" model. It is needed if "general-gamma" model is used.
K	integer greater than or equal to 2. It is needed if "K-gamma" model is used.
p	integer greater than or equal to 2. It is needed if "efron" mode is used.
c0	numeric and greater than 0. It is needed if "efron" mode is used.
maxi	upper limit of iteration for the ECM algorithm.
eps	a tolerance parameter used in the stopping rule of the ECM algorithm. If the difference in marginal likelihood between two consecutive iterations is less than eps, the ECM algorithm stops. Default to be 1e-4.
n_posterior_draws	number of posterior draws for each AE-drug combination.

Details

This function implements the ECM algorithm proposed by Tan et al. (2025), providing a stable and efficient implementation of Gamma-Poisson Shrinker(GPS), K-gamma and "general-gamma" methods for signal estimation and signal detection in Spontaneous Reporting System (SRS) data table.

Method "GPS" is proposed by DuMouchel (1999) and it is a parametric empirical Bayes model with a two gamma mixture prior distribution.

Methods "K-gamma" and "general-gamma" are non-parametric empirical Bayes models, introduced by Tan et al. (2025). The number of mixture components "K" needs to be prespecified when fitting a "K-gamma" model. For "general-gamma", the mixture weights are regularized by a Dirichlet hyper prior with hyperparameter $0 \leq \alpha < 1$ that controls the shrinkage strength. As "alpha" approaches 0, less non-empty mixture components exist in the fitted model. When $\alpha = 0$, the Dirichlet distribution is an improper prior still offering a reasonable posterior inference that represents the strongest shrinkage of the "general-gamma" model.

The implementation of the "KM" model relies on the **REBayes** package. The model fitting requires the MOSEK optimization solver. Please ensure that **Rmosek** is correctly installed and configured.

The implementation of the "efron" model in this package is adapted from the **deconvolveR** package, developed by Bradley Efron and Balasubramanian Narasimhan. The original implementation in **deconvolveR** does not support an exposure or offset parameter in the Poisson model, which corresponds to the expected null value (E_{ij}) for each AE-drug combination. To address this, we modified the relevant code to allow for the inclusion of E_{ij} in the Poisson likelihood. In addition, we implemented a method for estimating the degrees of freedom, enabling AIC- or BIC-based hyperparameter selection for the "efron" model (Tan et al. 2025). See `pvEBayes_tune` for details.

Value

The function returns an S3 object of class `pvEBayes` containing the estimated model parameters as well as posterior draws for each AE-drug combination if the number of posterior draws is specified.

References

DuMouchel W. Bayesian data mining in large frequency tables, with an application to the FDA spontaneous reporting system. *The American Statistician.* 1999; 1;53(3):177-90.

Tan Y, Markatou M and Chakraborty S. Flexible Empirical Bayesian Approaches to Pharmacovigilance for Simultaneous Signal Detection and Signal Strength Estimation in Spontaneous Reporting Systems Data. *arXiv preprint.* 2025; arXiv:2502.09816.

Narasimhan B, Efron B. deconvolveR: A G-modeling program for deconvolution and empirical Bayes estimation. *Journal of Statistical Software*. 2020; 2;94:1-20.

Koenker R, Gu J. REBayes: an R package for empirical Bayes mixture methods. *Journal of Statistical Software*. 2017; 4;82:1-26.

Examples

```
set.seed(1)
ref_table <- statin2025_44
```

```
# set up signal matrix with one signal at entry (1,1)
sig_mat <- matrix(1, nrow(ref_table), ncol(ref_table))
sig_mat[1, 1] <- 2

# set up structural zero matrix
Z <- matrix(0, nrow(ref_table), ncol(ref_table))
Z[5, 1] <- 1

simu_table <- generate_contin_table(
  ref_table = ref_table,
  signal_mat = sig_mat,
  n_table = 1,
  Variation = TRUE,
  zi_indic_mat = Z
)[[1]][[1]]


# fit general-gamma model with a specified alpha
fit <- pvEBayes(
  contin_table = simu_table, model = "general-gamma",
  alpha = 0.3, n_posterior_draws = 1000
)
```

`pvEBayes_tune` *Select hyperparameter and obtain the optimal general-gamma or efron model based on AIC and BIC*

Description

This function performs hyperparameter tuning for the general-gamma or Efron model using AIC or BIC. For a given AE-drug contingency table, the function fits a series of models across a grid of candidate hyperparameter values and computes their AIC and BIC. The models with the lowest AIC or BIC values are selected as the optimal fits.

Usage

```
pvEBayes_tune(
  contin_table,
  model = "general-gamma",
  alpha_vec = NULL,
  p_vec = NULL,
  c0_vec = NULL,
  use_AIC = TRUE,
  n_posterior_draws = 1000,
  return_all_fit = FALSE,
  return_all_AIC = TRUE,
  return_all_BIC = TRUE
)
```

Arguments

contin_table	an IxJ contingency table showing pairwise counts of adverse events for I AEs (along the rows) and J drugs (along the columns).
model	the model to be tuned. Available models are "general-gamma" and "efron". Default to "general-gamma".
alpha_vec	vector of hyperparameter alpha values to be selected. Alpha is a hyperparameter in general-gamma model which is numeric and between 0 and 1. If is NULL, a default set of alpha values (0, 0.1, 0.3, 0.5, 0.7, 0.9) will be used.
p_vec	vector of hyperparameter p values to be selected. p is a hyperparameter in "efron" model which should be a positive integer. If is NULL, a default set of p values (80, 100, 120, 150, 200) will be used.
c0_vec	vector of hyperparameter c0 values to be selected. c0 is a hyperparameter in "efron" model which should be a positive number. If is NULL, a default set of c0 values (0.001, 0.01, 0.1, 0.2, 0.5) will be used.
use_AIC	logical, indicating whether AIC or BIC is used. Default to be TRUE.
n_posterior_draws	number of posterior draws for each AE-drug combination.
return_all_fit	logical, indicating whether to return all the fitted model under the selection. Default to be FALSE.
return_all_AIC	logical, indicating whether to return AIC values for each fitted model under the selection. Default to be TRUE.
return_all_BIC	logical, indicating whether to return BIC values for each fitted model under the selection. Default to be TRUE.

Value

The function returns an S3 object of class `pvEBayes` containing the selected estimated model parameters as well as posterior draws for each AE-drug combination if the number of posterior draws is specified.

References

Akaike H. A new look at the statistical model identification. *IEEE Transactions on Automatic Control.* 2003; 19(6):716-23.

Schwarz G. Estimating the dimension of a model. *The Annals of Statistics.* 1978; 1:461-4.

Tan Y, Markatou M and Chakraborty S. Flexible Empirical Bayesian Approaches to Pharmacovigilance for Simultaneous Signal Detection and Signal Strength Estimation in Spontaneous Reporting Systems Data. *arXiv preprint.* 2025; arXiv:2502.09816.

Examples

```
fit <- pvEBayes_tune(statin2025_44, model = "general-gamma")
```

statin2025 *FDA statin dataset with 5119 adverse events*

Description

An adverse event-drug count dataset (contingency table) obtained from the FDA FAERS database for the quarters 2021Q1 - 2024Q4.

Usage

```
statin2025
```

Format

An object of class `matrix` (inherits from `array`) with 5119 rows and 7 columns.

Details

The dataset catalogs 6 statin drugs (across columns): Atorvastatin, Fluvastatin, Lovastatin, Pravastatin, Rosuvastatin, Simvastatin.

Data are stored in the form of a contingency table, with drugs listed across the columns and the 5119 AEs presented across the rows. Each cell in the contingency table represents the total report counts associated with that (AE, drug) pair and detected in the FDA FAERS database during 2021Q1 - 2024Q4.

The dataset catalogs 6 statin drugs (across columns):

Atorvastatin, Fluvastatin, Lovastatin, Pravastatin, Rosuvastatin, Simvastatin.

The 5119 adverse events presented across the rows are AEs that contain at least one report for 6 statin drugs during 2021Q1 - 2024Q4.

This dataset is an updated version of statin from the pvLRT package which collects the same scope of AEs for 6 statin drugs for quarters 2014Q3 - 2020Q4.

Source

https://fis.fda.gov/extensions/FPD-QDE-FAERS/FPD-QDE-FAERS.html

statin2025_44 *FDA statin dataset with 44 adverse events*

Description

An adverse event-drug count dataset (contingency table) obtained from the FDA FAERS database for the quarters 2021Q1 - 2024Q4.

Usage

```
statin2025_44
```

Format

An object of class `matrix` (inherits from `array`) with 45 rows and 7 columns.

Details

Data are stored in the form of a contingency table, with drugs listed across the columns and the 44 AEs presented across the rows. Each cell in the contingency table represents the total report counts associated with that (AE, drug) pair and detected in the FDA FAERS database during 2021Q1 - 2024Q4.

The dataset catalogs 6 statin drugs (across columns):

Atorvastatin, Fluvastatin, Lovastatin, Pravastatin, Rosuvastatin, Simvastatin.

The 44 adverse events presented across the rows are considered significant by FDA.

This dataset is an updated version of statin46 from the pvLRT package which collect the same scope of AEs for 6 statin drugs for quarters 2014Q3 - 2020Q4.

During 2021Q1 - 2024Q4, there was no AE report for "BLOOD CREATINE PHOSPHOKINASE MM INCREASED" and "MYOGLOBIN BLOOD PRESENT". Therefore, these two AEs are not presented in the statin2025_44 dataset.

Source

https://fis.fda.gov/extensions/FPD-QDE-FAERS/FPD-QDE-FAERS.html

---

statin42 *FDA statin dataset with 42 adverse events*

Description

An adverse event-drug count dataset (contingency table) obtained from the FDA FAERS database for the quarters 2014Q3 - 2020Q4.

Usage

```
statin42
```

Format

An object of class `matrix` (inherits from `array`) with 43 rows and 7 columns.

Details

Data are stored in the form of a contingency table, with drugs listed across the columns and the 42 AEs presented across the rows. Each cell in the contingency table represents the total report counts associated with that (AE, drug) pair and detected in the FDA FAERS database during 2014Q3 - 2020Q4.

The dataset catalogs 6 statin drugs (across columns):

Atorvastatin, Fluvastatin, Lovastatin, Pravastatin, Rosuvastatin, Simvastatin.

This dataset is derived from the `statin46` dataset in the **pvLRT** package, with four AEs removed.

The excluded AEs are: "Blood Creatine Phosphokinase Mm Increased", "Myoglobin Blood Present", "Myoglobin Urine Present", and "Myoglobinaemia".

Source

https://fis.fda.gov/extensions/FPD-QDE-FAERS/FPD-QDE-FAERS.html

---

`summary.pvEBayes` *Summary method for a pvEBayes object*

Description

This function defines the S3 `summary` method for objects of class `pvEBayes`. It provides a detailed summary of the fitted model.

Usage

```
## S3 method for class 'pvEBayes'
summary(object, return = NULL, ...)
```

Arguments

`object`	a `pvEBayes` object, which is the output of the function pvEBayes or pvEBayes_tune.
`return`	A character string specifying which component the summary function should return.Valid options include: "prior parameters", "likelihood", "detected signal" and "posterior draws". If set to NULL (default), all components will be returned in a list.
`...`	other input parameters. Currently unused.

Value

a list including estimated prior parameters, log_marginal_likelihood, indicator matrix of detected signal and posterior_draws for each AE-drug pair.

Examples

```
obj <- pvEBayes(
  contin_table = statin2025_44, model = "general-gamma",
  alpha = 0.5, n_posterior_draws = 10000
)

summary(obj)
```

Index