Package: pvm (via r-universe)

November 6, 2024

Title PharmacoVigilance Methods

Version 1.0

Author Louis Dijkstra [aut, cre], Marco Garling [ctb]

Maintainer Louis Dijkstra <dijkstra@leibniz-bips.de>

Description A collection of methods used in the field of pharmacovigilance for the dectection of 'interesting' drug-adverse event pairs from spontaneous reporting data.

License GPL (>= 3)

URL https://github.com/bips-hb/pvm

BugReports https://github.com/bips-hb/pvm/issues

Depends R (>= 3.2.3)

Imports glmnet, MCMCpack, Rcpp, stats

LinkingTo Rcpp

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Repository https://bips-hb.r-universe.dev

RemoteUrl https://github.com/bips-hb/pvm

RemoteRef HEAD

RemoteSha 44429e0422ed5cc8057acfb94732297172fe1efc

Contents

BCPNN *Bayesian Confidence Propagation Neural Network (BCPNN)*

Description

Applies the BCPNN to a collection of 2 x 2 tables of the form

	event	not event
drug	a	c
not drug	b	d

There are two versions of the BCPNN:

- `'original'` - The original version proposed by Bate et al. (1998)
- `'alternative'` - The BCPNN as proposed by Norén et al. (2006)

Usage

```
BCPNN(
  a,
  b,
  c,
  d,
  alpha = NULL,
  version = "original",
  mc_estimate = FALSE,
  mc_runs = 1000
)
```

Arguments

a	A vector with the counts of the upper left corner of the tables
b	A vector with the counts of the lower left corner of the tables
c	A vector with the counts of the upper right corner of the tables
d	A vector with the counts of the lower right corner of the tables
alpha	Value between $(0, 1)$. If set, the lower endpoint the confidence (or credible) interval is returned. (Default = NULL)
version	Version of the BCPNN used. Can either be 'original' (Default) for the BCPNN as proposed orignally by Bate et al. (1998), or 'alternative' for the BCPNN as proposed by Norén et al. (2006).
mc_estimate	The value is estimated using Monte Carlo runs (Default = FALSE). Only used when version = 'alternative'.
mc_runs	The number of Monte Carlo runs used to estimate the credible interval. (Default: 1000). Only used when version = 'alternative'.

Details

The implementation of this function is based on the implementation in the PhViD package.

Value

The maximum aposteriori estimate of the information component (IC) or the lower endpoint of the approximate credible interval

References

Bate, A., Lindquist, M., Edwards, I. R., Olsson, S., Orre, R., Lansner, A., & De Freitas, R. M. (1998). A Bayesian neural network method for adverse drug reaction signal generation. European Journal of Clinical Pharmacology, 54(4), 315–321. http://doi.org/10.1007/s002280050466

Norén, G. N., Bate, A., Orre, R., & Edwards, I. R. (2006). Extending the methods used to screen the WHO drug safety database towards analysis of complex associations and improved accuracy for rare events. Statistics in Medicine, 25(21), 3740–3757. http://doi.org/10.1002/sim.2473

Examples

```
# get the tables
a <- srdata$tables$a
b <- srdata$tables$b
c <- srdata$tables$c
d <- srdata$tables$d

# Applying the original BCPNN:
BCPNN(a, b, c, d)
# [1]  0.349783103 -0.609077730 -0.168446711 -0.277981964 ...
```

```
# Getting the lower end point of the 95% confidence intervaL:
BCPNN(a, b, c, d, alpha = 0.05)
# [1]  0.280077253 -0.994960076 -0.293624528 -0.408661852 ...

# Using the alternative version:
BCPNN(a, b, c, d, version = 'alternative')
# [1]  0.350235800 -0.595807902 -0.166901050 -0.276387348 ...

# Getting the lower end points of the 95% confidence interval
# using the alternative version. The estimates are based on
# 10,000 Monte Carlo samples:
BCPNN(a, b, c, d, version = 'alternative',
      alpha = 0.05, mc_estimate = TRUE, mc_runs = 10^4)
# [1]  [1]  0.31621489 -0.92490130 -0.25601307 -0.37040303 ...
```

`chi2Test` *Chi-squared Test*

Description

Performs the chi-squared test with or without Yates's continuity correction to a collection of 2 x 2 tables of the form

	event	not event
drug	a	c
not drug	b	d

Usage

```
chi2Test(a, b, c, d, yates = FALSE)
```

Arguments

a	A vector with the counts of the upper left corner of the tables
b	A vector with the counts of the lower left corner of the tables
c	A vector with the counts of the upper right corner of the tables
d	A vector with the counts of the lower right corner of the tables
yates	If TRUE, Yates's correction is used

Value

p-value

Note

The standard warnings for when the counts are too low in the 2 x 2 tables are suppressed. Due to the sparse nature of spontaneous reporting data, this happens quite frequently.

convertRawReports2Tables
Convert Reports to 2 x 2 Tables

Description

Creates a data frame containing all 2 x 2 contingency tables given a raw spontaneous reporting (SR) data set. An SR data set is a binary matrix, where each row is a report. The first columns represent the presence (1) or absence of a drug (0), the other columns represent the presence or absence of an event.

The tables are organized as follows:

	event j	not event j	*total*
drug i	a	c	a + c
not drug i	b	d	b + d
total	a + b	c + d	n_reports

Usage

```
convertRawReports2Tables(reports, n_drugs, n_events)
```

Arguments

reports	A binary matrix. Each row is a report
n_drugs	The number of drugs
n_events	The number of events

Details

The code is a simplified version of the function create2x2Tables in the SRSim package.

Value

A data frame where each row represents a 2 x 2 table. The columns represent:

drug_id	The ID of the drug
event_id	The ID of the event
a	Number of times the drug and event appeared together in a report
b	Number of times the event appeared without the drug in a report
c	Number of times the drug appeared without the event in a report
d	Number of times the drug and event both did not appear in a report

See Also

convertRawReports2Tables()

convertRawReports2TablesRcpp *Create 2 x 2 Tables*

Description

Creates a data frame containing all 2 x 2 contingency tables given a raw spontaneous reporting (SR) data set. An SR data set is a binary matrix, where each row is a report. The first columns represent the presence or absence of a drug, the The other columns represent the presence or absence of an event. See for more information the wrapper function, `convertRawReports2Tables()`.

Usage

```
convertRawReports2TablesRcpp(reports, n_drugs, n_events)
```

Arguments

`reports`	A binary matrix. Each row is a report
`n_drugs`	The number of drugs
`n_events`	The number of events

Details

The code is a simplified version of the function `create2x2TablesRcpp` in the `SRSim` package.

Value

A dataframe. A description of the columns can be found in the commentary for the function `convertRawReports2Tables()`

See Also

`convertRawReports2Tables()`

createTable *Create 2 x 2 Table*

Description

Returns a 2 x 2 contingency table of the form:

	event	not event
drug	a	c
not drug	b	d

Usage

```
createTable(a, b, c, d)
```

Arguments

a	Count in the upper left corner of the table
b	Count in the lower left corner of the table
c	Count in the upper right corner of the table
d	Count in the lower right corner of the table

Value

A 2 x 2 matrix

---

dbinbinom *Bimodal Negative Binomial*

Description

Returns the values of the density function of a bimodal negative binomial distribution.

Usage

```
dbinbinom(x, size1, prob1, size2, prob2, w)
```

Arguments

x	The x-values
size1, prob1	The size and prob parameters for the first mode
size2, prob2	The size and prob parameters for the second mode
w	The weight of the first mode (must lie in $[0, 1]$)

Value

The density for the values in x

References

DuMouchel, W. (1999). Bayesian Data Mining in Large Frequency Tables, with an Application to the FDA Spontaneous Reporting System. The American Statistician, 53(3), 177–190. https://doi.org/10.1080/00031305.1999.10474456

DuMouchel, W., & Pregibon, D. (2001). Empirical bayes screening for multi-item associations. Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '01, (October), 67–76. http://doi.org/10.1145/502512.502526

See Also

logLikelihood2NegativeBinomial(), fitPriorParametersGPS(), GPS()

---

fisherExactTest *Fisher's Exact Test*

Description

Performs the (one-sided) Fisher's exact test to a collection of 2 x 2 tables of the form:

	event	not event
drug	a	c
not drug	b	d

Usage

```
fisherExactTest(a, b, c, d, midpvalue = FALSE)
```

Arguments

a	A vector with the counts of the upper left corner of the tables
b	A vector with the counts of the lower left corner of the tables
c	A vector with the counts of the upper right corner of the tables
d	A vector with the counts of the lower right corner of the tables
midpvalue	The mid-p-value correction (suggested by Agresti) is applied

Details

Wrapper function for the Rcpp functions fishersTestGreater and midPFishersTestGreater.

Value

p-value

References

Ahmed, I., Dalmasso, C., Haramburu, F., Thiessard, F., Bro\"et, P., & Tubert-Bitter, P. (2010). False Discovery Rate Estimation for Frequentist Pharmacovigilance Signal Detection Methods. Biometrics, 66(1), 301–309. https://doi.org/10.1111/j.1541-0420.2009.01262.x

`fitPriorParametersGPS` *Prior Parameter Fit for the GPS*

Description

Fits the prior parameters to the data for the Gamma Poisson shrinker (GPS). The initial guess for the parameter values are set the same as by DuMouchel (1999).

Usage

```
fitPriorParametersGPS(
  a,
  b,
  c,
  d,
  E = ((a + b) * (a + c))/(a + b + c + d),
  alpha1 = 0.2,
  beta1 = 0.1,
  alpha2 = 2,
  beta2 = 4,
  w = 1/3
)
```

Arguments

`a`	A vector with the counts of the upper left corner of the tables
`b`	A vector with the counts of the lower left corner of the tables
`c`	A vector with the counts of the upper right corner of the tables
`d`	A vector with the counts of the lower right corner of the tables
`E`	Passed to `nlminb()` (Default = `((a + b)*(a + c)) / (a + b + c + d)`)
`alpha1`	Prior parameter α_1 (Default = 0.2)
`beta1`	Prior parameter β_1 (Default = 0.1)
`alpha2`	Prior parameter α_2 (Default = 2.0)
`beta2`	Prior parameter β_2 (Default = 4.0)
`w`	Prior parameter w (Default = 1/3)

Value

A list with the prior parameters

References

DuMouchel, W. (1999). Bayesian Data Mining in Large Frequency Tables, with an Application to the FDA Spontaneous Reporting System. The American Statistician, 53(3), 177–190. https://doi.org/10.1080/00031305.1999.10474456

DuMouchel, W., & Pregibon, D. (2001). Empirical bayes screening for multi-item associations. Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '01, (October), 67–76. http://doi.org/10.1145/502512.502526

See Also

`loglikelihood2NegativeBinomial()`

Examples

```
a <- srdata$tables$a
b <- srdata$tables$b
c <- srdata$tables$c
d <- srdata$tables$d

fitPriorParametersGPS(a, b, c, d)

# $alpha1
# [1] 98.28478
#
# $beta1
# [1] 16.48081
#
# $alpha2
# [1] 16.61439
#
# $beta2
# [1] 18.00642
#
# $w
# [1] 0.06132586
```

`GPS` *Gamma Poisson Shrinker (GPS)*

Description

Applies the Gamma Poisson Shrinker (GPS) introduced by DuMouchel (1999) to a collection of 2 x 2 tables of the form

	event	not event
drug	a	c
not drug	b	d

Usage

```
GPS(
  a,
  b,
  c,
  d,
  E = ((a + b) * (a + c))/(a + b + c + d),
  prior = fitPriorParametersGPS(a, b, c, d),
  alpha = NULL
)
```

Arguments

a	A vector with the counts of the upper left corner of the tables
b	A vector with the counts of the lower left corner of the tables
c	A vector with the counts of the upper right corner of the tables
d	A vector with the counts of the lower right corner of the tables
E	Vector with the expected values when there are no associations. By default set to the values used by DuMouchel (1999), i.e., `((a + b)*(a + c)) / (a + b + c + d)`.
prior	List that contains the prior parameters. If not specified, automatically fitted to the data, see `fitPriorParametersGPS()`.
alpha	Value between $(0, 1)$. If set, the lower endpoint the confidence (or credible) interval is returned. (Default = `NULL`)

Value

a vector with the GPS estimates

References

DuMouchel, W. (1999). Bayesian Data Mining in Large Frequency Tables, with an Application to the FDA Spontaneous Reporting System. The American Statistician, 53(3), 177–190. https://doi.org/10.1080/00031305.1999.10474456

DuMouchel, W., & Pregibon, D. (2001). Empirical bayes screening for multi-item associations. Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '01, (October), 67–76. http://doi.org/10.1145/502512.502526

See Also

`fitPriorParametersGPS()`

LASSO *LASSO*

Description

Applies the LASSO to raw spontaneous report data. Every event is regressed on all the drugs in the data set. The function returns a data frame with every drug-event pair and the estimated regression coefficient.

In case there are not enough observations of an event (the event must appear at least twice), the regression is not performed. All the regression estimates for the drugs and that particular event are set to `0`. The entries in the `lambda` column of the data frame are set to `NA`.

Shrinkage parameter

One can set the shrinkage parameter with the argument `lambda` in a number of ways:

1. `lambda = NULL` (*Default*). The parameter is set through cross-validation. The number of folds can be set with `nfolds` (Default = 10). The loss function used can be set with `type.measure` (Default = `deviance`). See for other `type.measure` options the function `glmnet::cv.glmnet`. The `glmnet::cv.glmnet` function returns two estimates: `lambda.min` and `lambda.1se`. To use the former, set `lambda.type` to `"min"` (default). For the latter, type `"1se"`.
2. Set to one value, e.g., `lambda = 0.5`. The same shrinkage parameter is used for all events.
3. A vector of length `n_events`, e.g., `lambda = c(0.5, 0.8, 1)`. The shrinkage parameters are specified for each event individually.

Usage

```
LASSO(
  reports,
  n_drugs,
  n_events,
  lambda = NULL,
  nfolds = 10,
  type.measure = "deviance",
  lambda.type = "min",
  alpha = 1,
  event_ids = 1:n_events,
  verbose = FALSE
)
```

Arguments

`reports`	A binary matrix, where each row represents a report
`n_drugs, n_events`	The number of drugs and events
`lambda`	Shrinkage parameter. Can be a list of length `n_events`. When not set, estimated through cross-validation

nfolds	Number of folds used for cross-validation
type.measure	Loss function used (Default: deviance). See for more options glmnet::cv.glmnet
lambda.type	Type of estimate that is used (either "min" - default - or "1se")
alpha	The elastic net mixing parameter (Default: 1.0 - LASSO)
event_ids	IDs of the events evaluated (Default: all)
verbose	Verbosity (Default: FALSE)

Value

A data frame with the columns

drug_id	ID for the drug (simply numbered 1,2,3,...etc.)
event_id	ID for the event (simply numbered 1,2,3,...etc.)
lambda	The shrinkage parameter λ that was used for this pair. In case the regression was not performed (because the event was not observed or only observed once), the entry is NA
LASSO	The regression parameter after regressing all drugs to the event in question

---

loglikelihood2NegativeBinomial

Log-Likelihood of the Bimodal Negative Binomial

Description

Returns the log-likelihood of the bimodal negative binomial model used by the Gamma Poisson shrinker (GPS()), see function fitPriorParametersGPS(). The function is written such that it can be used by the base function nlminb().

Usage

```
loglikelihood2NegativeBinomial(p, a, E)
```

Arguments

p	A vector with the parameters (alpha1, beta1, alpha2, beta2 and w, in that order)
a	A vector with the number of reports for each of the drug-event pairs
E	A vector (of the same length as a) with the number of reports one would expect under the assumption of 'independence'

Value

The negative log-likelihood (i.e., -1 * log-likelihood)

References

DuMouchel, W. (1999). Bayesian Data Mining in Large Frequency Tables, with an Application to the FDA Spontaneous Reporting System. The American Statistician, 53(3), 177–190. https://doi.org/10.1080/00031305.1999.10474456

DuMouchel, W., & Pregibon, D. (2001). Empirical bayes screening for multi-item associations. Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '01, (October), 67–76. http://doi.org/10.1145/502512.502526

See Also

GPS(), fitPriorParametersGPS(), dbinbinom()

Examples

```
alpha1 <- 0.2
beta1 <- 0.06
alpha2 <- 1.4
beta2 <- 1.8
w <- 0.1

a <- c(5, 1, 56, 3)
E <- c(3.4, 0.5, 10, 0.5)

p <- c(alpha1, beta1, alpha2, beta2, w)
loglikelihood2NegativeBinomial(p, a, E)
#[1] 16.80512
```

---

logLikelihoodRatioBinomial

Binomial Log-Likelihood Ratio Test

Description

Performs the log-likelihood ratio test for a collection of 2 x 2 tables of the form

	event	not event
drug	a	c
not drug	b	d

Usage

```
logLikelihoodRatioBinomial(a, b, c, d)
```

Arguments

a	A vector with the counts of the upper left corner of the tables
b	A vector with the counts of the lower left corner of the tables
c	A vector with the counts of the upper right corner of the tables
d	A vector with the counts of the lower right corner of the tables

Value

The log-likelihood ratio

References

Lian Duan, Khoshneshin, M., Street, W. N., & Mei Liu. (2013). Adverse drug effect detection. IEEE Journal of Biomedical and Health Informatics, 17(2), 305–11. https://doi.org/10.1109/TITB.2012.2227272

---

near *near*

Description

A safe way to compare two floating point numbers. The function is based on the near function in the dplyr package.

Usage

```
near(x, y, tol = .Machine$double.eps^0.5)
```

Arguments

x, y	Numeric vectors to compare
tol	Tolerance of comparison (Default: sqrt of the machine precision)

Value

TRUE when x and y are near, otherwise FALSE

PoissonTest *Test of the Poisson Mean*

Description

Performs the test of the Poisson mean to a collection of 2 x 2 tables of the form

	event	not event
drug	a	c
not drug	b	d

Usage

```
PoissonTest(a, b, c, d)
```

Arguments

a	A vector with the counts of the upper left corner of the tables
b	A vector with the counts of the lower left corner of the tables
c	A vector with the counts of the upper right corner of the tables
d	A vector with the counts of the lower right corner of the tables

Value

p-value

PRR *Proportional Reporting Rate (PRR)*

Description

Determines the proportional reporting rate to a collection of 2 x 2 tables of the form

	event	not event
drug	a	c
not drug	b	d

In case the parameter alpha is set, it returns the lower endpoint of the $100(1-\alpha)$ percent confidence interval.

Usage

```
PRR(a, b, c, d, alpha = NULL)
```

Arguments

a	A vector with the counts of the upper left corner of the tables
b	A vector with the counts of the lower left corner of the tables
c	A vector with the counts of the upper right corner of the tables
d	A vector with the counts of the lower right corner of the tables
alpha	Value between $(0, 1)$. If set, the lower endpoint the confidence (or credible) interval is returned. (Default = NULL)

Value

The PRR or the lower endpoint of the confidence interval

ROR *Reporting Odds Ratio (ROR)*

Description

Determines the ROR for a collection of 2 x 2 tables of the form:

	event	not event
drug	a	c
not drug	b	d

In case the parameter alpha is set, it returns the lower endpoint of the $100(1-\alpha)$ percent confidence interval.

Usage

```
ROR(a, b, c, d, alpha = NULL)
```

Arguments

a	A vector with the counts of the upper left corner of the tables
b	A vector with the counts of the lower left corner of the tables
c	A vector with the counts of the upper right corner of the tables
d	A vector with the counts of the lower right corner of the tables
alpha	Value between $(0, 1)$. If set, the lower endpoint the confidence (or credible) interval is returned. (Default = NULL)

Value

The ROR or the lower endpoint of the confidence interval of the ROR

RRR *Relative Reporting Risk (RRR)*

Description

Determines the RRR for a collection of 2 x 2 tables of the form

	event	not event
drug	a	c
not drug	b	d

Usage

```
RRR(a, b, c, d)
```

Arguments

a	A vector with the counts of the upper left corner of the tables
b	A vector with the counts of the lower left corner of the tables
c	A vector with the counts of the upper right corner of the tables
d	A vector with the counts of the lower right corner of the tables

Value

The RRR

srdata *A Simulated Spontaneous Reporting System*

Description

A simulated spontaneous reporting data set generated with the SRSim simulator. The data set contains 10,000 reports for 10 drugs and 10 adverse events (AEs). Five drug-AE pairs are associated with an odds ratio of 2. All other drug-AE pairs have an odds ratio of 1. Five drugs are innocent bystanders, i.e., they are prescribed together with one other drug, but they do not cause any adverse events.

Usage

```
srdata
```

Format

`srdata` contains the following elements:

sr A binary data frame with 10,000 rows and 20 columns. The first 10 columns represent the drugs; the latter represent the events. Each row is a report. In case of a 1, the drug/event has been reported, zero otherwise. The column names are `drug1` till `drug10` and `event1` till `event10`.

dag The directed acycled graph as an `igraph` object

nodes A tibble with all the information on each node/variate:

- `label` The label for each node/variate
- `in_degree` The number of edges pointing to the node
- `id` The ID of each node (simple integer)
- `parent_id` The ID of the parent node - if any. Otherwise equal to `-1`
- `margprob` The marginal probability of the node/variate
- `beta0` The intercept in the logistic regression model for that node
- `beta1` The regression coefficient in the logistic regression model for the parent

prob_drugs A vector with marginal probabilities of the drugs

prob_events A vector with marginal probabilities of the events

tables A data frame with 100 rows. Each row contains the data on a drug-event pair. The columns represent:

- `drug_id` The ID of the drug
- `event_id` The ID of the event
- `prob_drug` The marginal probability of that drug
- `prob_event` The marginal probability of that event
- `or` The odds ratio
- `associated` `TRUE` is there is a non-zero correlation, `FALSE` otherwise
- `a` Number of times the drug and event appeared together in a report
- `b` Number of times the event appeared without the drug in a report
- `c` Number of times the drug appeared without the event in a report
- `d` Number of times the drug and event both did not appear in a report

Details

The marginal probabilities over the drugs and the AEs were drawn from a Beta distribution with parameters $\alpha = 1.0$ and $\beta = 20.0$.

The conditional probability of an innocent bystander given that the other drug is prescribed is set to .9 (this is regulated with the argument `bystander_prob`).

The following commands were used for generating the data set:

```
library(SRSim)
srdata <- SRSim::simulateSRS(n_reports = 10000,
                             n_drugs = 10,
                             n_events = 10,
```

```
                              n_innocent_bystanders = 5,
                              bystander_prob = 0.9,
                              n_correlated_pairs = 5,
                              theta = 2,
                              seed = 1)
 # create the 2x2 tables

 srdata$tables <- SRSim::convert2Tables(srdata)
```

`YulesQ` *Yule's Q*

Description

Determines Yule's Q for a collection of 2 x 2 tables of the form

	event	not event
drug	a	c
not drug	b	d

In case the parameter `alpha` is set, it returns the lower endpoint of the $100(1-\alpha)$ percent confidence interval.

Usage

```
YulesQ(a, b, c, d, alpha = NULL)
```

Arguments

`a`	A vector with the counts of the upper left corner of the tables
`b`	A vector with the counts of the lower left corner of the tables
`c`	A vector with the counts of the upper right corner of the tables
`d`	A vector with the counts of the lower right corner of the tables
`alpha`	Value between $(0, 1)$. If set, the lower endpoint the confidence (or credible) interval is returned. (Default = `NULL`)

Value

Yule's Q or the lower endpoint of the confidence interval

Index