Package 'cusum'

October 12, 2022

Title Cumulative Sum (CUSUM) Charts for Monitoring of Hospital Performance

Version 0.4.1

Date 2019-10-02

Language en-GB

Description Provides functions for constructing and evaluating CUSUM charts and RA-CUSUM charts with focus on false signal probability.

Depends R (>= 3.5.0)

License GPL-2

LazyData true

SystemRequirements C++11

Encoding UTF-8

Imports stats, graphics, checkmate, data.table, Rcpp

Suggests testthat, knitr, ggplot2, rmarkdown

Enhances vdiffr, covr

LinkingTo Rcpp

VignetteBuilder knitr

RoxygenNote 6.1.1

NeedsCompilation yes

Author Lena Hubig [aut, cre] (<https://orcid.org/0000-0002-9454-1232>)

Maintainer Lena Hubig <cs@lenahubig.de>

Repository CRAN

Date/Publication 2019-10-02 09:30:03 UTC

R topics documented:

`calc_cusum` *Calculate CUSUM*

Description

This function calculates the CUSUM chart for the given sequence of successes and failures

Usage

```
calc_cusum(x, c0, cA)
```

Arguments

`x`	vector of outcomes
`c0`	accepted failure probability
`cA`	smallest detectable failure probability

Value

Returns matrix of possible sequences

cusum *Non-risk-adjusted CUSUM Charts*

Description

Calculate non-risk-adjusted CUSUM charts for performance data

Provides functions for constructing and evaluating CUSUM charts and RA-CUSUM charts with focus on false signal probability in health care processes.

Usage

```
cusum(failure_probability, patient_outcomes, limit, weights = NULL,
  odds_multiplier = 2, reset = TRUE)
```

Arguments

failure_probability	Double. Baseline failure probability
patient_outcomes	Integer. Vector of binary patient outcomes (0,1)
limit	Double. Control limit for signalling performance change
weights	Double. Optional vector of weights, if empty, standard CUSUM weights are calculated with weights_t
odds_multiplier	Double. Odds multiplier of adverse event under the alternative hypothesis (<1 looks for decreases)
reset	Logical. Reset the CUSUM after a signal to 0; defaults to TRUE

Author(s)

Lena Hubig

Examples

```
# control limit can be obtained with cusum_limit_sim(),
# here it is set to an arbitrary value (2.96)

# CUSUM of in-control process
# simulate patient outcomes
set.seed(2046)
patient_outcomes <- as.logical(rbinom(n = 100, size = 1, prob = 0.05))


cs_ic <- cusum(
  failure_probability = 0.05,
  patient_outcomes,
```

```
  limit = 2.96
)

# CUSUM of out-of-control process
# simulate patient outcome
set.seed(2046)
patient_outcomes <- as.logical(rbinom(n = 100, size = 1, prob = 0.2))

cs_oc <- cusum(
  failure_probability = 0.05,
  patient_outcomes,
  limit = 2.96
)
```

`cusum_alpha_sim` *Simulate false signal probability alpha given control limit for CUSUM charts*

Description

Simulate false signal probability alpha given control limit for CUSUM charts

Usage

```
cusum_alpha_sim(failure_probability, n_patients, odds_multiplier,
  n_simulation, limit, seed = NULL)
```

Arguments

`failure_probability`	Double. Baseline failure probability
`n_patients`	Integer. Number of patients in monitoring period /sample size
`odds_multiplier`	Double. Odds multiplier of adverse event under the alternative hypothesis (<1 looks for decreases)
`n_simulation`	Integer. Number of simulation runs
`limit`	Double. Control limit for signalling performance change
`seed`	Integer. Seed for RNG

Value

Returns False signal probability of specified CUSUM chart.

Examples

```
#
# control limit can be obtained with cusum_limit_sim(),
# here it is set to an arbitrary value (2.96)

# simulate false positive probability of CUSUM
cusum_alpha_sim(
  failure_probability = 0.05,
  n_patients = 100,
  odds_multiplier = 2,
  n_simulation = 10000,
  limit = 2.96,
  seed = 2046
)
```

`cusum_example_data` *Non-Risk-adjusted Performance Data*

Description

Generated performance data of indicator 17/1 54030: Preoperative stay over 24 hours for patients with proximal femur fracture.

Usage

```
cusum_example_data
```

Format

A data frame with 2000 rows and 3 variables:

t Sequence of observations

y Patient outcome

year Year of treatment

Details

Patient outcomes were simulated based on average national failure rate. Two years are provided, so Phase I and Phase II can be defined.

Source

Data for simulation was provided by Bavarian Agency of Quality Assurance (BAQ), Munich Germany.

Description of performance indicator (in German): https://iqtig.org/downloads/auswertung/2016/17n1hftfrak/QSKH_17n1-HUEFTFRAK_2016_QIDB_V02_2017-04-26.pdf

`cusum_limit_exact` *Calculate exact control limit given false signal probability alpha for CUSUM charts for very small sample sizes*

Description

This function only works for very small sample sizes (<= 15), as it permutes through all possible outcome sequences and estimates the percentage of runs that reach a specific CUSUM values.

Usage

```
cusum_limit_exact(n_patients, failure_probability, odds_multiplier, alpha)
```

Arguments

`n_patients`	Integer. Number of patients in monitoring period /sample size
`failure_probability`	Double. Baseline failure probability
`odds_multiplier`	Double. Odds multiplier of adverse event under the alternative hypothesis (<1 looks for decreases)
`alpha`	Double. False signal probability of CUSUM

Value

Returns the control limit for signalling performance change for small sample sizes (double)

Examples

```
# calculate exact control limits for alpha = 0.05
cusum_limit_exact(
  failure_probability = 0.1,
  n_patients = 10,
  odds_multiplier = 2,
  alpha = 0.05
)
```

`cusum_limit_sim` *Simulate control limit given false signal probability alpha for CUSUM charts*

Description

Simulate control limit given false signal probability alpha for CUSUM charts

Usage

```
cusum_limit_sim(failure_probability, n_patients, odds_multiplier,
  n_simulation, alpha, seed = NULL)
```

Arguments

`failure_probability`	Double. Baseline failure probability
`n_patients`	Integer. Number of patients in monitoring period /sample size
`odds_multiplier`	Double. Odds multiplier of adverse event under the alternative hypothesis (<1 looks for decreases)
`n_simulation`	Integer. Number of simulation runs
`alpha`	Double. False signal probability of CUSUM
`seed`	Integer. Seed for RNG

Value

Returns the control limit for signalling performance change (double)

Examples

```
# simulate control limits for alpha = 0.05
cusum_limit_sim(
  failure_probability = 0.05,
  n_patients = 100,
  odds_multiplier = 2,
  n_simulation = 1000,
  alpha = 0.05,
  seed = 2046
)
```

gscusum *Group-sequential CUSUM chart*

Description

Calculate GSCUSUM chart for non-risk-adjusted processes.

Usage

```
gscusum(input_outcomes, failure_probability, odds_multiplier, limit,
  quantiles, max_num_shuffles = 10000L, seed = 0L)
```

Arguments

input_outcomes	Matrix. First column are binary patient outcomes (0,1). Second column are continuous sequence of block identifier.
failure_probability	Double. Baseline failure probability
odds_multiplier	Double. Odds multiplier of adverse event under the alternative hypothesis (<1 looks for decreases)
limit	Double. Control limit for signalling performance change
quantiles	Double. Vector of requested quantiles of GSCUSUM distribution
max_num_shuffles	Integer. Number of shuffles (i.e. different sequences of observations)
seed	Integer. Seed for RNG (if = 0 random seed is set (default))

Value

gscusum matrix, signal probability, average CUSUM value and specified quantiles for every observation.

gscusum_example_data *Group-sequential Non-Risk-adjusted Performance Data with Block Identifier*

Description

Generated performance data of indicator 17/1 54030: Preoperative stay over 24 hours for patients with proximal femur fracture.

Usage

```
gscusum_example_data
```

Format

A data frame with 2000 rows and 4 variables:

t Sequence of observations

y Patient outcome

year Year of treatment

block_identifier Continuous block identifier

Details

Patient outcomes were simulated based on average national failure rate. Two years are provided, so Phase I and Phase II can be defined.

Source

Data for simulation was provided by Bavarian Agency of Quality Assurance (BAQ), Munich Germany.

Description of performance indicator (in German): https://iqtig.org/downloads/auswertung/2016/17n1hftfrak/QSKH_17n1-HUEFTFRAK_2016_QIDB_V02_2017-04-26.pdf

---

`make_all_outcomes` *Make all outcomes*

Description

creates all possible sequences of outcomes for a sample size

Usage

```
make_all_outcomes(npat_outcome)
```

Arguments

`npat_outcome`	Number of patients (sample sizes)

Value

Returns matrix of possible sequences

plot.cusum *Plot CUSUM chart for a cusum Object*

Description

Produces a CUSUM chart.

Usage

```
## S3 method for class 'cusum' plot(x, signal = TRUE, ...)
```

Arguments

x	An object of class cusum
signal	Logical. If TRUE, signals are plotted (default)

racusum *Risk-adjusted CUSUM Charts*

Description

Calculate risk-adjusted CUSUM charts for performance data

Usage

```
racusum(patient_risks, patient_outcomes, limit, weights = NULL,
  odds_multiplier = 2, reset = TRUE, limit_method = c("constant",
  "dynamic"))
```

Arguments

patient_risks	Double. Vector of patient risk scores (individual risk of adverse event)
patient_outcomes	Integer. Vector of binary patient outcomes (0,1)
limit	Double. Control limit for signalling performance change
weights	Double. Optional vector of weights, if empty, standard CUSUM weights are calculated with weights_t
odds_multiplier	Double. Odds multiplier of adverse event under the alternative hypothesis (<1 looks for decreases)
reset	Logical. Reset the CUSUM after a signal to 0; defaults to TRUE
limit_method	"constant" or "dynamic"

Examples

```
# Patients risks are usually known from Phase I.
# If not, these risk scores can be simulated.

# define possible patient risk scores
risks <- c(0.001, 0.01, 0.1, 0.002, 0.02, 0.2)

# sample risk population of size n = 100
set.seed(2046)
patient_risks <- sample(x = risks, size = 100, replace = TRUE)

# control limit can be obtained with racusum_limit_sim(),
# here it is set to an arbitrary value (2.96),
# or dynamic control limits with racusum_limit_dpcl()

##### RA-CUSUM of in-control process
# simulate patient outcome for performace as expected
set.seed(2046)
patient_outcomes <- as.logical(rbinom(
  n = 100,
  size = 1,
  prob = patient_risks
))

racusum(patient_risks,
  patient_outcomes,
  limit = 2.96
)

#### RA-CUSUM of out-of-control process
# simulate patient outcome for deviating performance

set.seed(2046)
patient_outcomes <- as.logical(rbinom(n = 100, size = 1, prob = patient_risks * 2))
#'
racusum(patient_risks,
  patient_outcomes,
  limit = 2.96
)
```

`racusum_alpha_sim` *Simulate false signal probability alpha given control limit for RA-CUSUM charts*

Description

Simulate false signal probability alpha given control limit for RA-CUSUM charts

Usage

```
racusum_alpha_sim(patient_risks, odds_multiplier, n_simulation, limit,
  seed = NULL)
```

Arguments

patient_risks	Double. Vector of patient risk scores (individual risk of adverse event)
odds_multiplier	Double. Odds multiplier of adverse event under the alternative hypothesis (<1 looks for decreases)
n_simulation	Integer. Number of simulation runs
limit	Double. Control limit for signalling performance change
seed	Integer. Seed for RNG

Value

Returns False signal probability of specified RA-CUSUM chart.

Examples

```
# Patients risks are usually known from Phase I.
# If not, these risk scores can be simulated.

# define possible patient risk scores
risks <- c(0.001, 0.01, 0.1, 0.002, 0.02, 0.2)

# sample risk population of size n = 100
set.seed(2046)
patient_risks <- sample(x = risks, size = 100, replace = TRUE)

# control limit can be obtained with racusum_limit_sim(),
# here it is set to an arbitrary value (2.96)

# simulate false positive probability of RA-CUSUM
racusum_alpha_sim(patient_risks,
  odds_multiplier = 2,
  n_simulation = 1000,
  limit = 2.96,
  seed = 2046
)
```

racusum_example_data *Risk-adjusted Performance Data*

Description

Generated performance data of indicator: Ratio of observed to expected cases of severe stroke or death under open carotid stenosis surgery.

Usage

```
racusum_example_data
```

Format

A data frame with 2000 rows and 4 variables:

t Sequence of observations

y Patient outcome

score Patient risk score

year Year of treatment

Details

Individual patient risk scores were drawn from actual hospital data and patient outcomes were simulated. Two years are provided, so Phase I and Phase II can be defined.

Source

Data for simulation was provided by Bavarian Agency of Quality Assurance (BAQ), Munich Germany.

Description of performance indicator (in German): https://iqtig.org/downloads/auswertung/2016/10n2karot/QSKH_10n2-KAROT_2016_QIDB_V02_2017-04-26.pdf

racusum_limit_dpcl *Dynamic Probability Control Limits (DPCL)*

Description

Set DPCL for risk-adjusted Bernoulli CUSUM Charts

Usage

```
racusum_limit_dpcl(patient_risks, N = 1e+05, odds_multiplier = 2,
  alpha, seed = NULL)
```

Arguments

patient_risks	Double. Vector of patient risk scores (individual risk of adverse event)
N	Integer. Number of simulation runs
odds_multiplier	Double. Odds multiplier of adverse event under the alternative hypothesis (<1 looks for decreases)
alpha	Double. False signal probability of RA-CUSUM
seed	Integer. Seed for RNG

Value

Returns vector of dynamic control limit for signalling performance change (double)

References

Zhang, Xiang & Woodall, William. (2016). Dynamic Probability Control Limits for Lower and Two-Sided Risk-Adjusted Bernoulli CUSUM Charts. Quality and Reliability Engineering International. 10.1002/qre.2044.

Examples

```
patient_risks <- runif(100, min = 0.1, max = 0.8)

dpcl <- racusum_limit_dpcl(
  patient_risks = patient_risks,
  N = 1000,
  odds_multiplier = 2,
  alpha = 0.05,
  seed = 32423
)

plot(dpcl, type = "l")
```

---

racusum_limit_sim *Simulate control limit given false signal probability alpha for RA-CUSUM charts*

Description

False-signal-probability-simulation of Control Limits h for risk-adjusted CUSUM charts

Usage

```
racusum_limit_sim(patient_risks, odds_multiplier, n_simulation, alpha,
  seed = NULL)
```

Arguments

patient_risks	Double. Vector of patient risk scores (individual risk of adverse event)
odds_multiplier	Double. Odds multiplier of adverse event under the alternative hypothesis (<1 looks for decreases)
n_simulation	Integer. Number of simulation runs
alpha	Double. False signal probability of RA-CUSUM
seed	Integer. Seed for RNG

Value

Returns the control limit for signalling performance change (double)

Examples

```
# Patients risks are usually known from Phase I.
# If not, these risk scores can be simulated.

# define possible patient risk scores
risks <- c(0.001, 0.01, 0.1, 0.002, 0.02, 0.2)

# sample risk population of size n = 100
set.seed(2046)
patient_risks <- sample(x = risks, size = 100, replace = TRUE)

# simulate control limits for alpha = 0.05
racusum_limit_sim(patient_risks,
  odds_multiplier = 2,
  n_simulation = 1000,
  alpha = 0.05,
  seed = 2046
)
```

---

ragscusum *RA-Grouped-CUSUM chart*

Description

Calculate GSCUSUM chart for risk-adjusted processes.

Usage

```
ragscusum(input_ra_outcomes, limit, quantiles, max_num_shuffles = 10000L,
  seed = 0L)
```

Arguments

`input_ra_outcomes`	Matrix. First column are binary patient outcomes (0,1). Second column are patient individual weight for adverse event (failure) and third column patient individual weight for no adverse event (success). Fourth column are continuous sequence of block identifier.
`limit`	Double. Control limit for signalling performance change
`quantiles`	Double. Vector of requested quantiles of RA-GSCUSUM distribution
`max_num_shuffles`	Integer. Number of shuffles (i.e. different sequences of observations)
`seed`	Integer. Seed for RNG (if = 0 random seed is set (default))

Value

ragscusum NumericMatix, signal probability, average CUSUM value and specified quantiles for every observation.

`ragscusum_example_data` *Group-sequential Risk-adjusted Performance Data with Block Identifier*

Description

Generated performance data of indicator: Ratio of observed to expected cases of severe stroke or death under open carotid stenosis surgery.

Usage

```
ragscusum_example_data
```

Format

A data frame with 2000 rows and 4 variables:

t Sequence of observations

y Patient outcome

score Patient risk score

year Year of treatment

block_identifier Continuous block identifier

Details

Individual patient risk scores were drawn from actual hospital data and patient outcomes were simulated. Two years are provided, so Phase I and Phase II can be defined.

Source

Data for simulation was provided by Bavarian Agency of Quality Assurance (BAQ), Munich Germany.

Description of performance indicator (in German): https://iqtig.org/downloads/auswertung/2016/10n2karot/QSKH_10n2-KAROT_2016_QIDB_V02_2017-04-26.pdf

---

`weights_t` *Weights for observations*

Description

Calculate standard CUSUM weights

Usage

```
weights_t(patient_outcomes, probability_ae, odds_multiplier = 2)
```

Arguments

`patient_outcomes`	Integer. Vector of binary patient outcomes (0,1)
`probability_ae`	Double. Baseline failure probability for adverse event in non-risk-adjusted case, vector of patient risk scores for risk-adjustment.
`odds_multiplier`	Double. Odds multiplier of adverse event under the alternative hypothesis (<1 looks for decreases)

Index