Package 'SNSchart'

April 7, 2021

Type Package

Title Sequential Normal Scores in Statistical Process Management

Version 1.4.0

Author Victor Tercero [aut],
Luis Benavides [aut, cre],
Jorge Merlo [ctb]

Maintainer Luis Benavides <luisbv@tec.mx>

Description The methods discussed in this package are new non-parametric methods based on sequential normal scores 'SNS' (Conover et al (2017) <doi:10.1080/07474946.2017.1360091>), designed for sequences of observations, usually time series data, which may occur singly or in batches, and may be univariate or multivariate. These methods are designed to detect changes in the process, which may occur as changes in location (mean or median), changes in scale (standard deviation, or variance), or other changes of interest in the distribution of the observations, over the time observed. They usually apply to large data sets, so computations need to be simple enough to be done in a reasonable time on a computer, and easily updated as each new observation (or batch of observations) becomes available. Some examples and more detail in 'SNS' is presented in the work by Conover et al (2019) <arXiv:1901.04443>.

Depends R (>= 2.10)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Imports parallel, stats, MASS

Suggests testthat, knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2021-04-07 08:40:15 UTC

R topics documented:

`calibrateControlLimit` *Calibration of the control limit for the selected chart*

Description

The methodology used to calibrate the control limit for the SNS chart depending on the selected chart

Usage

```
calibrateControlLimit(
  targetARL = NULL,
  targetMRL = NULL,
  n,
  m,
  theta = NULL,
  Ftheta = NULL,
  scoring = "Z",
  Chi2corrector = "None",
  dist,
  mu,
  sigma,
  dist.par = c(0, 1, 1),
  chart,
  chart.par,
  replicates = 50000,
  isParallel = TRUE,
  maxIter = 20,
  progress = TRUE,
  alignment = "unadjusted",
  constant = NULL,
  absolute = FALSE,
  isFixed = FALSE,
  rounding.factor = NULL
)
```

Arguments

`targetARL`	scalar. is the target ARL to calibrate. By default is set to NULL
`targetMRL`	scalar. is the target ARL to calibrate. By default is set to NULL
`n`	scalar. Subroup size
`m`	scalar. Reference sample size
`theta`	scalar. Value corresponig with the `Ftheta` quantile.
`Ftheta`	scalar. Quantile of the data distribution. The values that take are between (0,1).
`scoring`	character string. If "Z" (normal scores) (default). If "Z-SQ" (normal scores squared).
`Chi2corrector`	character string. Only when scoring is Z-SQ. Select from • "approx: Z^2*(m + 1 + 1.3)/(m+1). • "exact": Z^2/mean(Z). • "none": Z^2. If "approx" () (default). If "exact" (normal scores squared).
`dist`	character string. Select from: • "Uniform: Continuous Uniform distribution .

- "Normal": Normal distribution (default).
- "Normal2": Squared Normal distribution (also known as Chi-squared).
- "DoubleExp": Double exponential distribution (also known as Laplace distribution).
- "DoubleExp2": Double exponential squared distribution from a `DoubleExp(0,1)`.
- "LogNormal": Lognormal distribution.
- "Gamma": Gamma distribution.
- "Weibull": Weibull distribution.
- "t": Student-t distribution.

`mu` vector. Two elements, the first one is the mean of the reference sample and the second one is the mean of the monitoring sample.

`sigma` vector. Two elements, the first one is the sd of the reference sample and the second one is the sd of the monitoring sample.

`dist.par` vector. Distribution parameters. `c(par.a,par.b)`. Default `c(0,1)`.

`chart` character string. Selected type of chart. Three options are available: Shewhart, CUSUM, EWMA

`chart.par` vector. The size depends on the selected chart:

Shewhart scheme: is `c(k)`, where `k` comes from $UCL = mu + k\sigma, LCL = mu - k\sigma$.

CUSUM scheme: is `c(k,h,t)` where `k` is the reference value and `h` is the control limit, and `t` is the type of the chart (1:positive, 2:negative, 3:two sides)

EWMA scheme: is `c(lambda,L)`, where `lambda` is the smoothing constant and `L` multiplies standard deviation to get the control limit

`replicates` scalar. Number of replicates to get the ARL

`isParallel` logical. If `TRUE` the code runs in parallel according to the number of cores in the computer,otherwise the code runs sequentially. Default `TRUE`.

`maxIter` scalar. is a numeric. The maximum number of iteration to take the calibration before stops

`progress` logical. If `TRUE` it shows the progress in the console.

`alignment` character string. Aligment of the data `X` and `Y`. Select from

- "unadjusted": nothing is sustracte from `X` and `Y` (default).
- "overallmean": overall mean is sustracted from `X` and `Y`.
- "overallmedian": overall median is sustracted from `X` and `Y`.
- "samplemean": mean from corresponding group (`X` and `Y`) is sustracted from its corresponing vector.
- "samplemedian": median from corresponding group (`X` and `Y`) is sustracted from its corresponing vector.
- "referencemean": mean from `Y` is subtracted from `X` and `Y`.
- "referencemedian": median from `Y` is subtracted from `X` and `Y`.
- "constantvalue": a constant value is subtracted from `X` and `Y`.

`constant` scalar. Only used when the `alignment` is selected "constantvalue". Default `NULL`.

absolute	logical. If TRUE, the absolute aligned values are obtained. (Default FALSE)
isFixed	logical. If TRUE the reference sample does not update, otherwise the reference sample is updated whenever the batch is in control.
rounding.factor	scalar. positive value that determine the range between two consecutive rounded values.

Value

Multiple output. Select by output$

- objective.function: scalar. The best solution obtained, in terms of the target ARL or MRL
- par.value: scalar. Which parameter of the chart reach this best solution
- iter: scalar. In which iteration is found the objective function.
- found: boolean. Is TRUE if in the maxIter is reached the desired +-5

Note

The argument chart.par in this function correspond to the initial parameters to start the calibration.

Examples

```
n <- 2 # subgroup size
m <- 30 # reference-sample size
dist <- "Normal" # distribution
mu <- c(0, 0) # c(reference sample mean, monitoring sample mean)
sigma <- c(1, 1) # c(reference sample sd, monitoring sample sd)

#### Distribution parameters
dist.par <- c(0, 1) # c(location, scale)

#### Other Parameters
replicates <- 2
targetARL <- 370
isParallel = FALSE

#### Control chart parameters
chart <- "Shewhart"
chart.par <- c(3)
shewhart <- calibrateControlLimit(
  targetARL = targetARL, targetMRL = NULL, n = n, m = m, theta = NULL,
 Ftheta = NULL, dist = dist, mu = mu, sigma = sigma, dist.par = dist.par, chart.par = chart.par,
  replicates = replicates, chart = chart, isParallel = isParallel
)

chart <- "CUSUM"
chart.par <- c(0.5, 2.5, 3)
cusum <- calibrateControlLimit(
  targetARL = targetARL, targetMRL = NULL, n = n, m = m, theta = NULL,
 Ftheta = NULL, dist = dist, mu = mu, sigma = sigma, dist.par = dist.par, chart.par = chart.par,
```

```
  replicates = replicates, chart = chart, isParallel = isParallel
)

chart <- "EWMA"
chart.par <- c(0.2, 2.962)
ewma <- calibrateControlLimit(
  targetARL = targetARL, targetMRL = NULL, n = n, m = m, theta = NULL,
 Ftheta = NULL, dist = dist, mu = mu, sigma = sigma, dist.par = dist.par, chart.par = chart.par,
  replicates = replicates, chart = chart, isParallel = isParallel
)
```

`dataAlignment` *Alignment of the data*

Description

Align the monitoring sample `X` and the reference sample `Y`.

Usage

```
dataAlignment(
  X,
  Y,
  alignment = "unadjusted",
  constant = NULL,
  absolute = FALSE
)
```

Arguments

`X`	vector. Monitoring sample.
`Y`	vector. Reference sample.
`alignment`	character string. Aligment of the data `X` and `Y`. Select from • "unadjusted": nothing is sustracte from `X` and `Y` (default). • "overallmean": overall mean is sustracted from `X` and `Y`. • "overallmedian": overall median is sustracted from `X` and `Y`. • "samplemean": mean from corresponding group (`X` and `Y`) is sustracted from its corresponing vector. • "samplemedian": median from corresponding group (`X` and `Y`) is sustracted from its corresponing vector. • "referencemean": mean from `Y` is subtracted from `X` and `Y`. • "referencemedian": median from `Y` is subtracted from `X` and `Y`. • "constantvalue": a constant value is subtracted from `X` and `Y`.
`constant`	scalar. Only used when the `alignment` is selected "constantvalue". Default `NULL`.
`absolute`	logical. If `TRUE`, the absolute aligned values are obtained. (Default `FALSE`)

Value

Multiple output. Select by `output$`

- `X`: vector. Monitor sample with the alignment selected.
- `Y`: vector. Reference sample with the alignment selected.

Examples

```
X = c(30, 45, 50)
Y = c(20, 22, 25, 30, 70)
dataAlignment(X,Y)
```

`example49` *Data from Example 4.9 Qiu (2014).*

Description

A dataset containing the data set used in Example 4.9 of Qiu (2014).

Usage

```
example49
```

Format

A data frame with 50 rows and 6 columns:

Y1 Reference sample of the first data set. 10 batches are N(0,1)

X1 Monitoring sample of the first data set. 10 batches are N(1,1).

Y2 Reference sample of the second data set. 10 batches are N(0,1)

X2 Monitoring sample of the second data set. 10 batches are N(0,2^2).

X.id id of each observation of the batch for the second data set.

Source

https://users.phhp.ufl.edu/pqiu/research/book/spc/data/example49.dat

example65 *Data from Example 6.5 on page 246 Qiu (2014).*

Description

A dataset containing the data set used in Example 6.5 on page 246 of Qiu (2014).

Usage

```
example65
```

Format

A data frame with 30 rows and 5 columns:

x first 9 observations are the reference sample. Batch size equals to 1.
Wn Wn
Sn2 Sn2
Bmax Bmax
hn hn

Source

https://users.phhp.ufl.edu/pqiu/research/book/spc/data/example65.dat

example71 *Data from Example 7.1 Qiu (2014).*

Description

A dataset containing the data set used in Example 7.1 of Qiu (2014).

Usage

```
example71
```

Format

The data (X1,X2,X3) consist of 30 observations each variable.

X 1st batch.
X.1 2nd batch.
X.2 3rd batch.

Source

https://users.phhp.ufl.edu/pqiu/research/book/spc/data/example71.dat

example74a *Data from Example 7.4(a) Qiu (2014).*

Description

A dataset containing the data set used in Example 7.4(a) of Qiu (2014).

Usage

```
example74a
```

Format

The data (X1,X2,X3) consist of 30 observations each variable.

X 1st batch.

X.1 2nd batch.

X.2 3rd batch.

Source

https://users.phhp.ufl.edu/pqiu/research/book/spc/data/example74a.dat

example74b *Data from Example 7.4(b) Qiu (2014).*

Description

A dataset containing the data set used in Example 7.4(b) of Qiu (2014).

Usage

```
example74b
```

Format

The data (X1,X2,X3) consist of 30 observations each variable.

X 1st batch.

X.1 2nd batch.

X.2 3rd batch.

Source

https://users.phhp.ufl.edu/pqiu/research/book/spc/data/example74b.dat

example74c *Data from Example 7.4(c) Qiu (2014).*

Description

A dataset containing the data set used in Example 7.4(c) of Qiu (2014).

Usage

```
example74c
```

Format

The data (X1,X2,X3) consist of 30 observations each variable.

X 1st batch.

X.1 2nd batch.

X.2 3rd batch.

Source

https://users.phhp.ufl.edu/pqiu/research/book/spc/data/example74c.dat

example81 *Data from Example 8.1 on page 319 Qiu (2014).*

Description

A dataset containing the data set used in Example 8.1 on page 319 of Qiu (2014).

Usage

```
example81
```

Format

A data frame with 300 rows (30 batches of size equals to 10)

X observations of all batches

X.id id of each observation of the batch

Source

https://users.phhp.ufl.edu/pqiu/research/book/spc/data/example81.dat

example82 *Data from Example 8.2 Qiu (2014).*

Description

A dataset containing the data set used in Example 8.2 on page 323 of Qiu (2014).

Usage

```
example82
```

Format

A data frame with 150 rows (30 batches of size equals to 5)

X observations of all batches

X.id id of each observation of the batch

Source

https://users.phhp.ufl.edu/pqiu/research/book/spc/data/example82.dat

example83 *Data from Example 8.3 on page 326 Qiu (2014).*

Description

A dataset containing the data set used in Example 8.3 on page 326 of Qiu (2014).

Usage

```
example83
```

Format

A data frame with 180 rows (30 batches of size equals to 6)

X observations of all batches

X.id id of each observation of the batch

Source

https://users.phhp.ufl.edu/pqiu/research/book/spc/data/example81.dat

example84 *Data from Example 8.4 Qiu (2014).*

Description

A dataset containing the data set used in Example 8.4 of Qiu (2014).

Usage

```
example84
```

Format

A data frame with 150 rows (30 batches of size equals to 5)

X observations of all batches

X.id id of each observation of the batch

Source

https://users.phhp.ufl.edu/pqiu/research/book/spc/data/example84.dat

example85 *Data from Example 8.5 Qiu (2014).*

Description

A dataset containing the data set used in Example 8.5 of Qiu (2014).

Usage

```
example85
```

Format

A data frame with 300 rows (30 batches of size equals to 10)

X observations of all batches

X.id id of each observation of the batch

Source

https://users.phhp.ufl.edu/pqiu/research/book/spc/data/example85.dat

example87 *Data from Example 8.7 on page 339 Qiu (2014).*

Description

A dataset containing the data set used in Example 8.7 on page 339 of Qiu (2014).

Usage

```
example87
```

Format

A data frame with 86 rows (86 batches of size equals to 1)

X observations of all batches

X.id id of each observation of the batch

Y reference sample of size equals to 14

Source

https://users.phhp.ufl.edu/pqiu/research/book/spc/data/example87.dat

example91 *Data from Example 9.1 on page 369 Qiu (2014).*

Description

A dataset containing the data set used in Example 9.1 on page 369 of Qiu (2014).

Usage

```
example91
```

Format

The data (X,Y) consist of 20 batches with 50 observations in each batch.

V1 1st batch.

V2 2nd batch.

Source

https://users.phhp.ufl.edu/pqiu/research/book/spc/data/example91.dat

example93 *Data from Example 9.3 Qiu (2014).*

Description

A dataset containing the data set used in Example 9.3 of Qiu (2014).

Usage

```
example93
```

Format

The data (X,Y) consist of 20 batches with 10 observations in each batch.

X 1st batch.

X.1 2nd batch.

Source

https://users.phhp.ufl.edu/pqiu/research/book/spc/data/example93.dat

getARL *Average Run Length (ARL)*

Description

Get the ARL getRL

Usage

```
getARL(
  n,
  m,
  theta = NULL,
  Ftheta = NULL,
  dist,
  mu,
  sigma,
  dist.par = c(0, 1, 1),
  chart,
  chart.par,
  scoring = "Z",
  Chi2corrector = "None",
  replicates = 10000,
```

```
  isParallel = TRUE,
  print.RL = FALSE,
  progress = FALSE,
  calibrate = FALSE,
  arl0 = 370,
  alignment = "unadjusted",
  constant = NULL,
  absolute = FALSE,
  isFixed = FALSE,
  rounding.factor = NULL
)
```

Arguments

`n`	scalar. Subroup size
`m`	scalar. Reference sample size
`theta`	scalar. Value corresponig with the `Ftheta` quantile.
`Ftheta`	scalar. Quantile of the data distribution. The values that take are between (0,1).
`dist`	character string. Select from: • "Uniform: Continuous Uniform distribution . • "Normal": Normal distribution (default). • "Normal2": Squared Normal distribution (also known as Chi-squared). • "DoubleExp": Double exponential distribution (also known as Laplace distribution). • "DoubleExp2": Double exponential squared distribution from a `DoubleExp(0,1)`. • "LogNormal": Lognormal distribution. • "Gamma": Gamma distribution. • "Weibull": Weibull distribution. • "t": Student-t distribution.
`mu`	vector. Two elements, the first one is the mean of the reference sample and the second one is the mean of the monitoring sample.
`sigma`	vector. Two elements, the first one is the sd of the reference sample and the second one is the sd of the monitoring sample.
`dist.par`	vector. Distribution parameters. `c(par.a,par.b)`. Default `c(0,1)`.
`chart`	character string. Selected type of chart. Three options are available: Shewhart, CUSUM, EWMA
`chart.par`	vector. The size depends on the selected chart: **Shewhart scheme:** is `c(k)`, where `k` comes from $UCL = mu + k\sigma, LCL = mu - k\sigma$. **CUSUM scheme:** is `c(k,h,t)` where `k` is the reference value and `h` is the control limit, and `t` is the type of the chart (1:positive, 2:negative, 3:two sides) **EWMA scheme:** is `c(lambda,L)`, where `lambda` is the smoothing constant and `L` multiplies standard deviation to get the control limit

`scoring` character string. If "Z" (normal scores) (default). If "Z-SQ" (normal scores squared).

`Chi2corrector` character string. Only when scoring is Z-SQ. Select from

- "approx: Z^2*(m + 1 + 1.3)/(m+1).
- "exact": Z^2/mean(Z).
- "none": Z^2.

If "approx" () (default). If "exact" (normal scores squared).

`replicates` scalar. Number of replicates to get the ARL

`isParallel` logical. If `TRUE` the code runs in parallel according to the number of cores in the computer,otherwise the code runs sequentially. Default `TRUE`.

`print.RL` logical. If `TRUE` return the vectors of RL for each iteration.

`progress` logical. If `TRUE` it shows the progress in the console.

`calibrate` logical. If `TRUE` the RL is limit to 10 times the target ARL.

`arl0` scalar. Expected value of the RL. Default `370`.

`alignment` character string. Aligment of the data `X` and `Y`. Select from

- "unadjusted": nothing is sustracte from `X` and `Y` (default).
- "overallmean": overall mean is sustracted from `X` and `Y`.
- "overallmedian": overall median is sustracted from `X` and `Y`.
- "samplemean": mean from corresponding group (`X` and `Y`) is sustracted from its corresponing vector.
- "samplemedian": median from corresponding group (`X` and `Y`) is sustracted from its corresponing vector.
- "referencemean": mean from `Y` is subtracted from `X` and `Y`.
- "referencemedian": median from `Y` is subtracted from `X` and `Y`.
- "constantvalue": a constant value is subtracted from `X` and `Y`.

`constant` scalar. Only used when the `alignment` is selected "constantvalue". Default `NULL`.

`absolute` logical. If `TRUE`, the absolute aligned values are obtained. (Default `FALSE`)

`isFixed` logical. If `TRUE` the reference sample does not update, otherwise the reference sample is updated whenever the batch is in control.

`rounding.factor`
scalar. positive value that determine the range between two consecutive rounded values.

Value

Multiple output. Select by `output$`

- `ARL`: scalar. Average Run Length for the `RL`s of all the `replicates`.
- `SDRL`: scalar. Standard Deviation Run Length for the `RL` in all the `replicates`.
- `MRL`: bolean. Median Run Length for the `RL`s of all the `replicates`.
- `QRL`: vector. It retrieve the quantiles (0.05, 0.1, 0.2, 0.25, 0.5, 0.75, 0.8, 0.9, 0.95) for all the `RL`s.

Examples

```
n <- 5 # subgroup size
m <- 100 # reference-sample size
dist <- "Normal"
mu <- c(0, 0) # c(reference sample mean, monitoring sample mean)
sigma <- c(1, 1) # c(reference sample sd, monitoring sample sd)

#### Normal distribution parameters
dist.par <- c(0, 1) # c(location, scale)

#### Other Parameters
replicates <- 2
print.RL <- TRUE
isParallel <- FALSE
calibrate <- FALSE
progress <- TRUE
arl0 <- 370

#### Control chart parameters
chart <- "Shewhart"
chart.par <- c(3)
shewhart <- getARL(n, m,
  theta = NULL, Ftheta = NULL, dist, mu, sigma, dist.par = dist.par,
  chart = chart, chart.par = chart.par, print.RL = print.RL,
  replicates = replicates, isParallel = isParallel,
  calibrate = calibrate, arl0 = arl0
)

chart <- "CUSUM"
chart.par <- c(0.25, 4.4181, 3)
cusum <- getARL(n, m,
  theta = NULL, Ftheta = NULL, dist, mu, sigma, dist.par = dist.par,
  chart = chart, chart.par = chart.par, print.RL = print.RL,
  replicates = replicates, isParallel = isParallel,
  calibrate = calibrate, arl0 = arl0
)

chart <- "EWMA"
chart.par <- c(0.2, 2.962)
shewhart <- getARL(n, m,
  theta = NULL, Ftheta = NULL, dist, mu, sigma, dist.par = dist.par,
  chart = chart, chart.par = chart.par, print.RL = print.RL,
  replicates = replicates, isParallel = isParallel,
  calibrate = calibrate, arl0 = arl0
)
```

`getDist` *Random Observations Generator*

Description

Random observations generator selected from several distributions with user defined mean and variance.

Usage

```
getDist(
  n,
  dist,
  mu,
  sigma,
  par.location = 0,
  par.scale = 1,
  par.shape = 1,
  dist.par = NULL,
  rounding.factor = NULL
)
```

Arguments

`n`	scalar. Number of observations to be generated.
`dist`	character string. Select from: • "Uniform: Continuous Uniform distribution . • "Normal": Normal distribution (default). • "Normal2": Squared Normal distribution (also known as Chi-squared). • "DoubleExp": Double exponential distribution (also known as Laplace distribution). • "DoubleExp2": Double exponential squared distribution from a `DoubleExp(0,1)`. • "LogNormal": Lognormal distribution. • "Gamma": Gamma distribution. • "Weibull": Weibull distribution. • "t": Student-t distribution.
`mu`	scalar. Expected value of the desired distribution.
`sigma`	scalar. Standard deviation of the desired distribution.
`par.location`	scalar. Location parameter of the desired distribution. Default 0**.
`par.scale`	scalar. Scale parameter of the desired distribution. Default 1**.
`par.shape`	scalar. Shape parameter of the desired distribution, Default 1.
`dist.par`	vector. Overwrite `par.location`, `par.scale`, `par.shape`. Depends on the distribution (default `NULL`): • "Uniform: no matter how is defined always gives numbers between 0 and 1. • "Normal": c(location, scale). • "Normal2": c(location, scale). • "DoubleExp": c(location, scale).

- "DoubleExp2": c(location, scale).
- "LogNormal": c(location, scale).
- "Gamma": c(scale, shape).
- "Weibull": c(shape, scale).
- "t": c(degrees of freedom).

`rounding.factor`
: scalar. positive value that determine the range between two consecutive rounded values.

Value

A vector `x` with `n` observations generated following the selected distribution with its parameters.

**Note

- For "Lognormal", `par.location` and `par.scale` correspond to the location and scale parameters of the normal distribution that generales the lognormal. Hence, in this case they are the logmean and the logsigma parameters
- For "Normal2" and "DoubleExp2", `par.location` and `par.scale` correspond correspond to the location and scale parameters of the normal and double exponential that are used to generates their squared forms.

Examples

```
getDist(1, "Normal", 0, 1)
```

---

getQuantile *Obtain Quantile from Distribution Function*

Description

Get the quantile `theta` from several distributions with user defined mean and variance.

Usage

```
getQuantile(
  Ftheta,
  mu,
  sigma,
  dist,
  par.location = 0,
  par.scale = 1,
  par.shape = 1,
  dist.par = NULL
)
```

Arguments

`Ftheta`	scalar. Quantile of the data distribution. The values that take are between (0,1).
`mu`	scalar. Expected value of the desired distribution.
`sigma`	scalar. Standard deviation of the desired distribution.
`dist`	character string. Select from: • "Uniform: Continuous Uniform distribution . • "Normal": Normal distribution (default). • "Normal2": Squared Normal distribution (also known as Chi-squared). • "DoubleExp": Double exponential distribution (also known as Laplace distribution). • "DoubleExp2": Double exponential squared distribution from a `DoubleExp(0,1)`. • "LogNormal": Lognormal distribution. • "Gamma": Gamma distribution. • "Weibull": Weibull distribution. • "t": Student-t distribution.
`par.location`	scalar. Location parameter of the desired distribution. Default 0**.
`par.scale`	scalar. Scale parameter of the desired distribution. Default 1**.
`par.shape`	scalar. Shape parameter of the desired distribution, Default 1.
`dist.par`	vector. Overwrite `par.location`, `par.scale`, `par.shape`. Depends on the distribution (default `NULL`): • "Uniform: no matter how is defined always gives numbers between 0 and 1. • "Normal": c(location, scale). • "Normal2": c(location, scale). • "DoubleExp": c(location, scale). • "DoubleExp2": c(location, scale). • "LogNormal": c(location, scale). • "Gamma": c(scale, shape). • "Weibull": c(shape, scale). • "t": c(degrees of freedom).

Value

A quantile `theta` of the selected `Ftheta` distribution with its parameters.

Examples

```
getQuantile(0.5, 0, 1, "Normal")
```

getRL *Run Length*

Description

Get the run length

Usage

```
getRL(
  replica = 1,
  n,
  m,
  theta = NULL,
  Ftheta = NULL,
  dist,
  mu,
  sigma,
  dist.par = c(0, 1, 1),
  scoring = "Z",
  chart,
  chart.par,
  calibrate = FALSE,
  arl0 = 370,
  alignment = "unadjusted",
  constant = NULL,
  absolute = FALSE,
  isFixed = FALSE,
  Chi2corrector = "None",
  rounding.factor = NULL
)
```

Arguments

replica	scalar. It is used for the parallel version of the function (parallel=TRUE). Default 1.
n	scalar. Subroup size
m	scalar. Reference sample size
theta	scalar. Value corresponig with the Ftheta quantile.
Ftheta	scalar. Quantile of the data distribution. The values that take are between (0,1).
dist	character string. Select from: • "Uniform: Continuous Uniform distribution . • "Normal": Normal distribution (default). • "Normal2": Squared Normal distribution (also known as Chi-squared).

- "DoubleExp": Double exponential distribution (also known as Laplace distribution).
- "DoubleExp2": Double exponential squared distribution from a `DoubleExp(0,1)`.
- "LogNormal": Lognormal distribution.
- "Gamma": Gamma distribution.
- "Weibull": Weibull distribution.
- "t": Student-t distribution.

`mu` vector. Two elements, the first one is the mean of the reference sample and the second one is the mean of the monitoring sample.

`sigma` vector. Two elements, the first one is the sd of the reference sample and the second one is the sd of the monitoring sample.

`dist.par` vector. Distribution parameters. `c(par.a,par.b)`. Default `c(0,1)`.

`scoring` character string. If "Z" (normal scores) (default). If "Z-SQ" (normal scores squared).

`chart` character string. Selected type of chart. Three options are available: Shewhart, CUSUM, EWMA

`chart.par` vector. The size depends on the selected chart:

Shewhart scheme: is `c(k)`, where `k` comes from $UCL = mu + k\sigma, LCL = mu - k\sigma$.

CUSUM scheme: is `c(k,h,t)` where `k` is the reference value and `h` is the control limit, and `t` is the type of the chart (1:positive, 2:negative, 3:two sides)

EWMA scheme: is `c(lambda,L)`, where `lambda` is the smoothing constant and `L` multiplies standard deviation to get the control limit

`calibrate` logical. If `TRUE` the RL is limit to 10 times the target ARL.

`arl0` scalar. Expected value of the RL. Default `370`.

`alignment` character string. Aligment of the data `X` and `Y`. Select from

- "unadjusted": nothing is sustracte from `X` and `Y` (default).
- "overallmean": overall mean is sustracted from `X` and `Y`.
- "overallmedian": overall median is sustracted from `X` and `Y`.
- "samplemean": mean from corresponding group (`X` and `Y`) is sustracted from its corresponing vector.
- "samplemedian": median from corresponding group (`X` and `Y`) is sustracted from its corresponing vector.
- "referencemean": mean from `Y` is subtracted from `X` and `Y`.
- "referencemedian": median from `Y` is subtracted from `X` and `Y`.
- "constantvalue": a constant value is subtracted from `X` and `Y`.

`constant` scalar. Only used when the `alignment` is selected "constantvalue". Default `NULL`.

`absolute` logical. If `TRUE`, the absolute aligned values are obtained. (Default `FALSE`)

`isFixed` logical. If `TRUE` the reference sample does not update, otherwise the reference sample is updated whenever the batch is in control.

`Chi2corrector` character string. Only when scoring is Z-SQ. Select from

- "approx: Z^2*(m + 1 + 1.3)/(m+1).
- "exact": Z^2/mean(Z).
- "none": Z^2.

If "approx" () (default). If "exact" (normal scores squared).

`rounding.factor`

scalar. positive value that determine the range between two consecutive rounded values.

Value

RL vector. The run length of the chart for the parameter setting.

Examples

```
n <- 5 # subgroup size
m <- 100 # reference-sample size
dist <- "Normal"
mu <- c(0, 0) # c(reference sample mean, monitoring sample mean)
sigma <- c(1, 1) # c(reference sample sd, monitoring sample sd)

#### Distribution parameters
dist.par <- c(0, 1, 1) # c(location, scale, shape)

#### Other Parameters
replicates <- 2
print.RL <- TRUE
calibrate <- FALSE
progress <- TRUE
arl0 <- 370

#### Control chart parameters
chart <- "Shewhart"
chart.par <- c(3)
shewhart <- getRL(1, n, m,
  theta = NULL, Ftheta = NULL,dist, mu, sigma, dist.par = dist.par,
  chart = chart, chart.par = chart.par, calibrate = calibrate, arl0 = arl0
)

chart <- "CUSUM"
chart.par <- c(0.25, 4.4181, 3)
cusum <- getRL(1, n, m,
  theta = NULL, Ftheta = NULL, dist, mu, sigma, dist.par = dist.par,
  chart = chart, chart.par = chart.par, calibrate = calibrate, arl0 = arl0
)

chart <- "EWMA"
chart.par <- c(0.2, 2.962)
shewhart <- getRL(1, n, m,
  theta = NULL, Ftheta = NULL,dist, mu, sigma, dist.par = dist.par,
  chart = chart, chart.par = chart.par, calibrate = calibrate, arl0 = arl0
)
```

`mcalibrateControlLimit` *Calibration of the control limit for the selected chart*

Description

The methodology used to calibrate the control limit for the SNS chart depending on the selected chart

Usage

```
mcalibrateControlLimit(
  targetARL = NULL,
  targetMRL = NULL,
  n,
  m,
  nv,
  theta = NULL,
  Ftheta = NULL,
  dists = c("Normal", "Normal"),
  mu = c(0, 0),
  sigma = NULL,
  dists.par = matrix(c(0, 1, 1, 0, 1, 1), ncol = 2),
  correlation = 0,
  chart = "T2",
  chart.par = c(10),
  replicates = 50000,
  isParallel = FALSE,
  maxIter = 20,
  progress = TRUE,
  alignment = "unadjusted",
  constant = NULL,
  absolute = FALSE
)
```

Arguments

`targetARL`	scalar. is the target ARL to calibrate. By default is set to NULL
`targetMRL`	scalar. is the target ARL to calibrate. By default is set to NULL
`n`	scalar. Subroup size
`m`	scalar. Reference sample size
`nv`	scalar. Number of variables to be generated.
`theta`	vector. Value corresponding with the `Ftheta` quantile.
`Ftheta`	vector. Quantile of the data distribution. The values that take are between (0,1).
`dists`	list. Select the

mu	vector. Two elements of the vector the first one is the mean of the reference sample and the second one is the mean of the monitoring sample.
sigma	scalar. Standard deviation of the desired distribution.
dists.par	matrix For each variable (column), specify • par.location: Location parameter of the desired distribution. Default 0. • par.scale: Scale parameter of the desired distribution. Default 1. • par.shape: Shape parameter of the desired distribution, Default 1. The number of columns must be the same as the number of variables.
correlation	scalar. Corralation between variables.
chart	character string. Selected type of chart. One option available: "T2". **T2 scheme:** is c(k), where k comes from $UCL = mu+k\sigma, LCL = mu-k\sigma$.
chart.par	vector. Control limit and other parameters of the selected chart.
replicates	scalar. Number of replicates to get the ARL
isParallel	logical. If TRUE the code runs in parallel according to the number of cores in the computer,otherwise the code runs sequentially. Default TRUE.
maxIter	scalar. is a numeric. The maximum number of iteration to take the calibration before stops
progress	logical. If TRUE it shows the progress in the console.
alignment	character string. Aligment of the data X and Y. Select from • "unadjusted": nothing is sustracte from X and Y (default). • "overallmean": overall mean is sustracted from X and Y. • "overallmedian": overall median is sustracted from X and Y. • "samplemean": mean from corresponding group (X and Y) is sustracted from its corresponing vector. • "samplemedian": median from corresponding group (X and Y) is sustracted from its corresponing vector. • "referencemean": mean from Y is subtracted from X and Y. • "referencemedian": median from Y is subtracted from X and Y. • "constantvalue": a constant value is subtracted from X and Y.
constant	scalar. Only used when the alignment is selected "constantvalue". Default NULL.
absolute	logical. If TRUE, the absolute aligned values are obtained. (Default FALSE)

Value

Multiple output. Select by output$

- objective.function: scalar. The best solution obtained, in terms of the target ARL or MRL
- par.value: scalar. Which parameter of the chart reach this best solution
- found: boolean. Is TRUE if in the maxIter is reached the desired +-5

Note

The argument `chart.par` in this function correspond to the initial parameters to start the calibration.

Examples

```
n <- 5 # subgroup size
m <- 10 # reference-sample size
dists <- c("Normal", "Normal") # distribution
mu <- c(0, 0) # c(reference sample mean, monitoring sample mean)
nv <- 2 # number of variables
#### Other Parameters
replicates <- 2
targetARL <- 200
isParallel = FALSE
maxIter <- 2
#### Control chart parameters
chart <- "T2"
chart.par <- c(0.005)
t2 <- mcalibrateControlLimit(targetARL = targetARL,n = n, m = m, nv = nv, theta = NULL,
  Ftheta = NULL, dists = dists, mu = mu, chart.par = chart.par,
  replicates = replicates, chart = chart, isParallel = isParallel,
  maxIter = maxIter
)
```

`mgetARL` *Multivariate Average Run Length (ARL)*

Description

Get the ARL `getRL`

Usage

```
mgetARL(
  n,
  m,
  nv,
  theta = NULL,
  Ftheta = NULL,
  dists,
  dists.par = NULL,
  mu,
  sigma = NULL,
  chart = "T2",
  chart.par = c(0.005),
  correlation = 0,
  s = NULL,
```

```
  replicates = 10000,
  isParallel = TRUE,
  print.RL = FALSE,
  progress = FALSE,
  calibrate = FALSE,
  arl0 = 370,
  alignment = "unadjusted",
  constant = NULL,
  absolute = FALSE
)
```

Arguments

n	scalar. Subroup size
m	scalar. Reference sample size
nv	scalar. Number of variables to be generated.
theta	vector. Value corresponding with the Ftheta quantile.
Ftheta	vector. Quantile of the data distribution. The values that take are between (0,1).
dists	list. Select the
dists.par	matrix For each variable (column), specify • par.location: Location parameter of the desired distribution. Default 0. • par.scale: Scale parameter of the desired distribution. Default 1. • par.shape: Shape parameter of the desired distribution, Default 1. The number of columns must be the same as the number of variables.
mu	vector. Two elements of the vector the first one is the mean of the reference sample and the second one is the mean of the monitoring sample.
sigma	scalar. Standard deviation of the desired distribution.
chart	character string. Selected type of chart. One option available: "T2". **T2 scheme:** is c(k), where k comes from $UCL = mu+k\sigma, LCL = mu-k\sigma$.
chart.par	vector. Control limit and other parameters of the selected chart.
correlation	scalar. Corralation between variables.
s	matrix. Correlation matrix of the variables
replicates	scalar. Number of replicates to get the ARL
isParallel	logical. If TRUE the code runs in parallel according to the number of cores in the computer,otherwise the code runs sequentially. Default TRUE.
print.RL	logical. If TRUE return the vectors of RL for each iteration.
progress	logical. If TRUE it shows the progress in the console.
calibrate	logical. If TRUE the RL is limit to 10 times the target ARL.
arl0	scalar. Expected value of the RL. It is only used for stop the RL if exceeds 10 times its value. Default 370.
alignment	character string. Aligment of the data X and Y. Select from

- "unadjusted": nothing is sustracte from X and Y (default).
- "overallmean": overall mean is sustracted from X and Y.
- "overallmedian": overall median is sustracted from X and Y.
- "samplemean": mean from corresponding group (X and Y) is sustracted from its corresponing vector.
- "samplemedian": median from corresponding group (X and Y) is sustracted from its corresponing vector.
- "referencemean": mean from Y is subtracted from X and Y.
- "referencemedian": median from Y is subtracted from X and Y.
- "constantvalue": a constant value is subtracted from X and Y.

constant — scalar. Only used when the alignment is selected "constantvalue". Default NULL.

absolute — logical. If TRUE, the absolute aligned values are obtained. (Default FALSE)

Value

Multiple output. Select by output$

- ARL: scalar. Average Run Length for the RLs of all the replicates.
- SDRL: scalar. Standard Deviation Run Length for the RL in all the replicates.
- MRL: bolean. Median Run Length for the RLs of all the replicates.
- QRL: vector. It retrieve the quantiles (0.05, 0.1, 0.2, 0.25, 0.5, 0.75, 0.8, 0.9, 0.95) for all the RLs.

Examples

```
mgetARL(replicates=5,n=5,m=100,nv=2,mu=c(0,0),
dists = c("Normal", "Normal"), dists.par = matrix(c(0,1,1,0,1,1), ncol=2),
isParallel=FALSE)
```

---

mgetDist *Multivariate Random Observations Generetor*

Description

Multivariate Random observations generator selected from several distributions with user defined mean and variance.

Usage

```
mgetDist(
  n,
  nv,
  mu = 0,
  sigma = NULL,
```

```
  correlation = 0,
  s = NULL,
  dists = NULL,
  dists.par = NULL
)
```

Arguments

n	scalar. Number of observations to be generated.
nv	scalar. Number of variables to be generated.
mu	scalar. Expected value of the desired distribution.
sigma	scalar. Standard deviation of the desired distribution.
correlation	scalar. Corralation between variables.
s	matrix. Correlation matrix of the variables
dists	list. Select the
dists.par	matrix For each variable (column), specify • par.location: Location parameter of the desired distribution. Default 0. • par.scale: Scale parameter of the desired distribution. Default 1. • par.shape: Shape parameter of the desired distribution, Default 1. The number of columns must be the same as the number of variables.

Value

A matrix x with n observations generated following the selected distribution with its parameters.

Examples

```
mgetDist(n=5, nv=2, dists=c("Normal", "Normal"),dists.par= matrix(c(0,1,1,0,1,1), ncol=2))
```

---

mgetRL *Multivariate Run Length*

Description

Get the run length

Usage

```
mgetRL(
  replica = 1,
  n,
  m,
  nv,
  theta = NULL,
  Ftheta = NULL,
```

```
  dists,
  mu,
  sigma = NULL,
  dists.par = NULL,
  correlation = 0,
  s = NULL,
  chart = "T2",
  chart.par = c(0.005),
  null.dist = "Chi",
  alignment = "unadjusted",
  constant = NULL,
  absolute = FALSE,
  calibrate = FALSE,
  arl0 = 370
)
```

Arguments

replica	scalar. It is used for the parallel version of the function (parallel=TRUE). Default 1.
n	scalar. Subroup size
m	scalar. Reference sample size
nv	scalar. Number of variables to be generated.
theta	vector. Value corresponding with the Ftheta quantile.
Ftheta	vector. Quantile of the data distribution. The values that take are between (0,1).
dists	list. Select the
mu	vector. Two elements of the vector the first one is the mean of the reference sample and the second one is the mean of the monitoring sample.
sigma	scalar. Standard deviation of the desired distribution.
dists.par	matrix For each variable (column), specify • par.location: Location parameter of the desired distribution. Default 0. • par.scale: Scale parameter of the desired distribution. Default 1. • par.shape: Shape parameter of the desired distribution, Default 1. The number of columns must be the same as the number of variables.
correlation	scalar. Corralation between variables.
s	matrix. Correlation matrix of the variables
chart	character string. Selected type of chart. One option available: "T2". **T2 scheme:** is c(k), where k comes from $UCL = mu+k\sigma, LCL = mu-k\sigma$.
chart.par	vector. Control limit and other parameters of the selected chart.
null.dist	character string. It is the null distribution choose from "Chi" or "F".
alignment	character string. Aligment of the data X and Y. Select from • "unadjusted": nothing is sustracte from X and Y (default).

- "overallmean": overall mean is sustracted from X and Y.
- "overallmedian": overall median is sustracted from X and Y.
- "samplemean": mean from corresponding group (X and Y) is sustracted from its corresponing vector.
- "samplemedian": median from corresponding group (X and Y) is sustracted from its corresponing vector.
- "referencemean": mean from Y is subtracted from X and Y.
- "referencemedian": median from Y is subtracted from X and Y.
- "constantvalue": a constant value is subtracted from X and Y.

constant — scalar. Only used when the alignment is selected "constantvalue". Default NULL.

absolute — logical. If TRUE, the absolute aligned values are obtained. (Default FALSE)

calibrate — logical. If TRUE the RL is limit to 10 times the target ARL.

arl0 — scalar. Expected value of the RL. It is only used for stop the RL if exceeds 10 times its value. Default 370.

Value

RL vector. The run length of the chart for the parameter setting.

Examples

```
mgetRL(n=5, m=10, nv=2, mu=c(0,0), dists = c("Normal", "Normal"),
dists.par = matrix(c(0,1,1,0,1,1), ncol=2))
```

MNS *Multivariate Normal Scores*

Description

Get conditional or unconditional multivariate normal score (NS) of observations (X) relative to previous observations (Y).

Usage

```
MNS(
  X,
  Y = NULL,
  theta = NULL,
  Ftheta = NULL,
  scoring = "Z",
  alignment = "unadjusted",
  constant = NULL,
  absolute = FALSE
)
```

Arguments

X	matrix or data.frame. New observations to obtain the normal scores.
Y	matrix or data.frame. If Y is not defined (no previous observation available, NULL), NS is relative to X. Default NULL.
theta	vector. Value corresponding with the Ftheta quantile.
Ftheta	vector. Quantile of the data distribution. The values that take are between (0,1).
scoring	character string. If "Z" (normal scores) (default). If "Z-SQ" (normal scores squared).
alignment	character string. Aligment of the data X and Y. Select from • "unadjusted": nothing is sustracte from X and Y (default). • "overallmean": overall mean is sustracted from X and Y. • "overallmedian": overall median is sustracted from X and Y. • "samplemean": mean from corresponding group (X and Y) is sustracted from its corresponing vector. • "samplemedian": median from corresponding group (X and Y) is sustracted from its corresponing vector. • "referencemean": mean from Y is subtracted from X and Y. • "referencemedian": median from Y is subtracted from X and Y. • "constantvalue": a constant value is subtracted from X and Y.
constant	scalar. Only used when the alignment is selected "constantvalue". Default NULL.
absolute	logical. If TRUE, the absolute aligned values are obtained. (Default FALSE)

Value

Multiple output. Select by output$

- R: matrix. Multivariate Ranks for the X observations. If ties occurs, average ranks are used.
- P: matrix. Multivariate Probability of the ranks for the X observations. Instead of Van Der Waerden normal scores where $P = R/(n+1)$, $P = (R-0.5)/n$, where R stands for rank and P for the input evaluated in the inverse of a Standard Normal Distribution.
- Z: matrix. Multivariate Normal scores for the X observations. Z if scoring is "Z" and Z^2 if scoring is "Z-SQ".

Examples

```
Y <- c(10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100)
Y = matrix(Y, ncol=2)
X <- c(30, 35, 45, 30, 35, 45)
X = matrix(X, ncol=2)
theta <- c(40, 40)
Ftheta <- c(0.5, 0.5)
# EXAMPLE CONDITIONAL
MNS(X = X, Y = Y, theta = theta, Ftheta = Ftheta)
```

MSNS *Multivariate Sequential Normal Scores*

Description

Transform a matrix X into SNS using initial observations Y if available SNS follow the order of X.

Usage

```
MSNS(
  X,
  X.id,
  Y = NULL,
  theta = NULL,
  Ftheta = NULL,
  scoring = "Z",
  alignment = "unadjusted",
  constant = NULL,
  absolute = FALSE,
  chart = "T2",
  chart.par = c(0.005),
  null.dist = "Chi",
  isFixed = FALSE,
  omit.id = NULL,
  auto.omit.alarm = TRUE
)
```

Arguments

X	matrix or data.frame. New observations to obtain the normal scores.
X.id	vector. The id of each column (variable) of the matrix X.
Y	matrix or data.frame. If Y is not defined (no previous observation available, NULL), NS is relative to X. Default NULL.
theta	vector. Value corresponding with the Ftheta quantile.
Ftheta	vector. Quantile of the data distribution. The values that take are between (0,1).
scoring	character string. If "Z" (normal scores) (default). If "Z-SQ" (normal scores squared).
alignment	character string. Aligment of the data X and Y. Select from • "unadjusted": nothing is sustracte from X and Y (default). • "overallmean": overall mean is sustracted from X and Y. • "overallmedian": overall median is sustracted from X and Y. • "samplemean": mean from corresponding group (X and Y) is sustracted from its corresponing vector.

- "samplemedian": median from corresponding group (X and Y) is sustracted from its corresponing vector.
- "referencemean": mean from Y is subtracted from X and Y.
- "referencemedian": median from Y is subtracted from X and Y.
- "constantvalue": a constant value is subtracted from X and Y.

`constant` scalar. Only used when the `alignment` is selected "constantvalue". Default `NULL`.

`absolute` logical. If `TRUE`, the absolute aligned values are obtained. (Default `FALSE`)

`chart` character string. Selected type of chart. One option available: `"T2"`.

T2 scheme: is `c(k)`, where `k` comes from $UCL = mu+k\sigma, LCL = mu-k\sigma$.

`chart.par` vector. Control limit and other parameters of the selected chart.

`null.dist` character string. It is the null distribution choose from `"Chi"` or `"F"`.

`isFixed` logical. If `TRUE` the reference sample does not update, otherwise the reference sample is updated when the batch is in control.

`omit.id` vector. Elements of the vector are the id which are omitted in the analysis.

`auto.omit.alarm` logical. Determine if OC signals are added (or not) to reference sample. By default is set to TRUE.

Value

Multiple output. Select by `output$`

- `coefficients`: list. Two elements: `n` the number of observation per group in `X` and `chart` the selected chart to perform the analysis.
- `X`: vector. New observations (Monitoring sample) to obtain the SNS.
- `Z`: vector. SNS of the `X` monitoring sample.
- `T2`: vector. T2 statistic for each of the groups in `X`.
- `X.id`: vector. The id of each column (variable) of the matrix `X`.
- `UCL`: vector. Upper control limit for each group in `X`.

Comments

If ties, average ranks are used.

See Also

`MNS` for multivariate normal scores

Examples

```
X = cbind(example91$X1, example91$X2)
X.id = example91$X.id
msns = MSNS(X, X.id)
```

NS *Normal Scores*

Description

Get conditional or unconditional normal score (NS) of observations (X) relative to previous observations (Y).

Usage

```
NS(
  X,
  Y = NULL,
  theta = NULL,
  Ftheta = NULL,
  scoring = "Z",
  Chi2corrector = "None",
  alignment = "unadjusted",
  constant = NULL,
  absolute = FALSE
)
```

Arguments

X	vector. New observations to obtain the N¡normal scores.
Y	vector. If Y is not defined (no previous observation available, NULL), NS is relative to X. Default NULL.
theta	scalar. Value corresponig with the Ftheta quantile.
Ftheta	scalar. Quantile of the data distribution. The values that take are between (0,1).
scoring	character string. If "Z" (normal scores) (default). If "Z-SQ" (normal scores squared).
Chi2corrector	character string. Only when scoring is Z-SQ. Select from • "approx: Z^2*(m + 1 + 1.3)/(m+1). • "exact": Z^2/mean(Z). • "none": Z^2. If "approx" () (default). If "exact" (normal scores squared).
alignment	character string. Aligment of the data X and Y. Select from • "unadjusted": nothing is sustracte from X and Y (default). • "overallmean": overall mean is sustracted from X and Y. • "overallmedian": overall median is sustracted from X and Y. • "samplemean": mean from corresponding group (X and Y) is sustracted from its corresponing vector. • "samplemedian": median from corresponding group (X and Y) is sustracted from its corresponing vector.

- "referencemean": mean from Y is subtracted from X and Y.
- "referencemedian": median from Y is subtracted from X and Y.
- "constantvalue": a constant value is subtracted from X and Y.

constant — scalar. Only used when the alignment is selected "constantvalue". Default NULL.

absolute — logical. If TRUE, the absolute aligned values are obtained. (Default FALSE)

Value

Multiple output. Select by output$

- R: vector. Ranks for the X observations. If ties occurs, average ranks are used.
- P: vector. Probability of the ranks for the X observations. Instead of Van Der Waerden normal scores where $P = R/(n+1)$, $P = (R-0.5)/n$, where R stands for rank and P for the input evaluated in the inverse of a Standard Normal Distribution.
- Z: vector. Normal scores for the X observations. Z if scoring is "Z" and Z^2 if scoring is "Z-SQ".

Examples

```
Y <- c(10, 20, 30, 40, 50, 60, 70, 80, 90, 100)
X <- c(30, 35, 45)
theta <- 40
Ftheta <- 0.5
# EXAMPLE CONDITIONAL
NS(X = X, Y = Y, theta = theta, Ftheta = Ftheta)

# EXAMPLE UNCONDITIONAL
theta <- NULL
Ftheta <- NULL
NS(X = X, Y = Y, theta = theta, Ftheta = Ftheta)
```

SNS *Sequential Normal Scores*

Description

Transform a vector X into SNS using initial observations Y if available SNS follow the order of X.

Usage

```
SNS(
  X,
  X.id,
  Y = NULL,
  theta = NULL,
  Ftheta = NULL,
```

```
  scoring = "Z",
  Chi2corrector = "None",
  alignment = "unadjusted",
  constant = NULL,
  absolute = FALSE,
  chart = "Shewhart",
  chart.par = c(3),
  snsRaw = FALSE,
  isFixed = FALSE,
  omit.id = NULL,
  auto.omit.alarm = TRUE
)
```

Arguments

X	vector. New observations to obtain the N¡normal scores.
X.id	vector. The id of the vector X.
Y	vector. If Y is not defined (no previous observation available, NULL), NS is relative to X. Default NULL.
theta	scalar. Value corresponig with the Ftheta quantile.
Ftheta	scalar. Quantile of the data distribution. The values that take are between (0,1).
scoring	character string. If "Z" (normal scores) (default). If "Z-SQ" (normal scores squared).
Chi2corrector	character string. Only when scoring is Z-SQ. Select from • "approx: Z^2*(m + 1 + 1.3)/(m+1). • "exact": Z^2/mean(Z). • "none": Z^2. If "approx" () (default). If "exact" (normal scores squared).
alignment	character string. Aligment of the data X and Y. Select from • "unadjusted": nothing is sustracte from X and Y (default). • "overallmean": overall mean is sustracted from X and Y. • "overallmedian": overall median is sustracted from X and Y. • "samplemean": mean from corresponding group (X and Y) is sustracted from its corresponing vector. • "samplemedian": median from corresponding group (X and Y) is sustracted from its corresponing vector. • "referencemean": mean from Y is subtracted from X and Y. • "referencemedian": median from Y is subtracted from X and Y. • "constantvalue": a constant value is subtracted from X and Y.
constant	scalar. Only used when the alignment is selected "constantvalue". Default NULL.
absolute	logical. If TRUE, the absolute aligned values are obtained. (Default FALSE)
chart	character string. Selected type of chart. Three options are available: Shewhart, CUSUM, EWMA

`chart.par` vector. The size depends on the selected chart:

Shewhart scheme: is `c(k)`, where `k` comes from $UCL = mu + k\sigma, LCL = mu - k\sigma$.

CUSUM scheme: is `c(k,h,t)` where `k` is the reference value and `h` is the control limit, and `t` is the type of the chart (1:positive, 2:negative, 3:two sides)

EWMA scheme: is `c(lambda,L)`, where `lambda` is the smoothing constant and `L` multiplies standard deviation to get the control limit

`snsRaw` logical. If `TRUE` return also the sns for each observation in vector `X`.

`isFixed` logical. If `TRUE` the reference sample does not update, otherwise the reference sample is updated whenever the batch is in control.

`omit.id` vector. Elements of the vector are the id which are omitted in the analysis.

`auto.omit.alarm` logical. Determine if OC signals are added (or not) to reference sample. By default is set to TRUE.

Value

Multiple output. Select by `output$`

- `coefficients`: list. Three elements: `n` the number of observation per group in `X`, `chart` the selected chart to perform the analysis, and `chart.par` the parameters of the selected chart.
- `R`: vector. Ranks for the new observations (Monitoring sample).
- `X`: vector. New observations (Monitoring sample) to obtain the SNS.
- `Z`: vector. SNS of the `X` monitoring sample.
- `X.id`: vector. The id of each column (variable) of the matrix `X`.
- `UCL`: vector. Upper control limit for each group in `X`.
- `LCL`: vector. Lower control limit for each group in `X`.
- `scoring`: string. Selected score to evaluate SNS.

Comments

If ties occur, average ranks are used.

See Also

`NS` for normal scores

Examples

```
# EXAMPLE CONDITIONAL WITH REFERENCE SAMPLE
Y <- c(10, 20, 30, 40, 50, 60, 70, 80, 90, 100)
X <- c(30, 35, 45)
theta <- 40
Ftheta <- 0.5
sample.id <- c("a", "b", "c")
SNS(X = X, X.id = sample.id, Y = Y, theta = theta, Ftheta = Ftheta)
```

```
# EXAMPLE CONDITIONAL WITH REFERENCE SAMPLE
Y <- c(10, 20, 30, 40, 50, 60, 70, 80, 90, 100)
X <- c(30, 35, 45)
theta <- 40
Ftheta <- 0.5
sample.id <- c("a", "b", "c")
SNS(X = X, X.id = sample.id, Y = Y, theta = theta, Ftheta = Ftheta)

# EXAMPLE UNCONDITIONAL WITH REFERENCE SAMPLE
Y <- c(10, 20, 30, 40, 50, 60, 70, 80, 90, 100)
X <- c(30, 35, 45)
theta <- NULL
Ftheta <- NULL
sample.id <- c("a", "b", "c")
SNS(X = X, X.id = sample.id, Y = Y, theta = theta, Ftheta = Ftheta)

# EXAMPLE CONDITIONAL WITHOUT REFERENCE SAMPLE
Y <- NULL # c(10,20,30,40,50,60,70,80,90,100)
X <- c(30, 35, 45)
theta <- 40
Ftheta <- 0.5
sample.id <- c("a", "b", "c")
SNS(X = X, X.id = sample.id, Y = Y, theta = theta, Ftheta = Ftheta)

# EXAMPLE UNCONDITIONAL WITHOUT REFERENCE SAMPLE
Y <- NULL
X <- c(30, 35, 45)
theta <- NULL
Ftheta <- NULL
sample.id <- c("a", "b", "c")
SNS(X = X, X.id = sample.id, Y = Y, theta = theta, Ftheta = Ftheta)
```

`srank` *Sequential Rank*

Description

Get the sequential rank of observations (`X`) relative to previous observations (`Y`).

Usage

```
srank(X, Y = NULL)
```

Arguments

`X`	vector. New observations to obtain the N¡normal scores.
`Y`	vector. If `Y` is not defined (no previous observation available, `NULL`), NS is relative to `X`. Default `NULL`.

Value

vector. Sequentil Ranks for the X observations. If ties occurs, average of the ranks are used.

Examples

```
X <- c(30, 35, 45)
srank(X)
```

Index