

Package: IQCC (via r-universe)

October 14, 2026

Title Improved Quality Control Charts

Version 0.8.0

Description Implements improved statistical process control charts with exact, Cornish-Fisher corrected, standardized, and simulation-based probability limits. Methods include univariate attribute and variable charts, double-sampling charts for rare nonconformities, Hotelling T-squared charts, and generalized variance charts for multivariate process variability. Numerical functions for limits, false-alarm risk, average run length, and average sample size are separated from plotting interfaces to support reproducible statistical validation.

Depends R (>= 3.5.0)

License GPL-2

URL <https://github.com/flaviobarros/IQCC>,
<https://flaviobarros.github.io/IQCC/>

BugReports <https://github.com/flaviobarros/IQCC/issues>

Encoding UTF-8

LazyData true

Language en-US

RoxygenNote 7.3.2

Imports MASS, miscTools, qcc

Suggests covr, knitr, rmarkdown, spelling, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

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

Date/Publication 2026-08-15 19:31:43 UTC

RemoteUrl <https://github.com/flaviobarros/iqcc>

RemoteRef HEAD

RemoteSha 69a1d018375d1625837b39bda124e2598195c6e4

Contents

IQCC-package	3
add.data	4
alpha.risk	5
binomdata	6
c4	7
cchart.DSnp	8
cchart.GV	10
cchart.p	14
cchart.R	15
cchart.S	17
cchart.T2.1	18
cchart.T2.2	19
cchart.trV	21
cchart.u	23
cchart.Xbar	24
cchart.Xbar_R	25
cchart.Xbar1	26
cchart.Xbar2	27
d2	28
d3	29
data.1	30
data.2	31
dsnp_ar1	32
dsnp_ass	33
dsnp_design	35
dsnp_limits	37
dsnp_prob_accept	39
gv_alpha_risk	41
gv_limits	43
gv_stat	46
moonroof	47
pchart_alpha_risk	48
pchart_limits	49
pistonrings	50
plot.cchart.DSnp	51
print.cchart.DSnp	51
print.cchart.GV	52
r_exact_limits	53
r_shewhart_limits	55
remove.data	56
s_exact_limits	57
s_shewhart_limits	58
sim_t2_asymptotic	60
softdrink	62
stats	63
T2.1	64

T2.2	65
table.const	66
table.qtukey	67
trv_alpha_risk	68
trv_limits	70
trv_stat	71
uchart_alpha_risk	72
uchart_limits	73

Index	75
--------------	-----------

IQCC-package	<i>IQCC: Improved Quality Control Charts</i>
--------------	--

Description

IQCC implements statistical process control charts with exact, Cornish-Fisher corrected, standardized, or simulation-based probability limits. The package covers univariate variable and attribute charts, double-sampling np charts for rare nonconformities, Hotelling T-squared charts, and generalized variance charts for multivariate variability.

Details

Numerical functions for limits, false-alarm risk, average run length, and average sample size are exposed separately from plotting functions so that calculations can be inspected, tested, and reused in simulation studies.

Useful links

- <https://github.com/flaviobarros/IQCC>
- <https://flaviobarros.github.io/IQCC/>
- Report bugs at <https://github.com/flaviobarros/IQCC/issues>

Author(s)

Maintainer: Flavio Barros <flaviomargarito@gmail.com>

Other contributors:

- Emanuel Barbosa [contributor]
- Elias Goncalves [contributor]
- Daniela Recchia [contributor]

See Also

[pchart_limits](#), [uchart_limits](#), [dsnp_limits](#), [gv_limits](#), [cchart.R](#), [cchart.DSnp](#), [cchart.GV](#)

add.data

Add a New Observation to a Phase II T^2 Chart

Description

Add a new Phase II point to an existing Hotelling T^2 chart. Computes the T^2 statistic via [T2.2](#) and plots it. Use after [cchart.T2.2](#) to build the chart incrementally.

Usage

```
add.data(datum2, estat, T2II, n, j, m = NULL)
```

Arguments

datum2	New Phase II data (vector for $n = 1$, matrix for $n > 1$).
estat	Phase I statistics (list from stats).
T2II	Previous T^2 vector (updated in calling code).
n	Subgroup size. Use $n = 1$ for individual observations, $n > 1$ for subgroups.
j	Index of this new sample.
m	Optional. Number of Phase I samples (needed if Phase I data is shown).

Details

To use this function it is necessary to have the output given by the function [T2.2](#). At every step you should enter the new data set.

Value

Invisibly, the new T^2 statistic.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

See Also

[T2.2](#), [cchart.T2.2](#)

Examples

```
mu <- c(5.682, 88.22)
Sigma <- miscTools::symMatrix(c(3.770, -5.495, 13.53), 2)
datum <- data.1(20, 10, mu, Sigma)
estat <- stats(datum, 20, 10, 2)
datum2 <- data.2(estat, 10, p = 2)
T2II <- T2.2(datum2, estat, 10)
#Not showing the phase I data set.
```

```

cchart.T2.2(T2II, 20, 10, 1, 25, 2)
datum3 <- data.2(estat, 10, p = 2)
add.data(datum3, estat, T2II, 10, 2)
#Showing the phase I data set.
cchart.T2.2(T2II, 20, 10, 1, 25, 2, datum = datum)
datum3 <- data.2(estat, 10, p = 2)
add.data(datum3, estat, T2II, 10, 2, 20)

#Example with individual observations
datum <- data.1(50, 1, mu, Sigma)
estat <- stats(datum, 50, 1, 2)
datum2 <- data.2(estat, 1, p = 2)
T2II <- T2.2(datum2, estat, 1)
#Not showing the phase I data set.
cchart.T2.2(T2II, 50, 1, 1, 25, 2)
datum3 <- data.2(estat, 1, p = 2)
add.data(datum3, estat, T2II, 1, 2)
#Showing the phase I data set.
cchart.T2.2(T2II, 50, 1, 1, 25, 2, datum = datum)
datum3 <- data.2(estat, 1, p = 2)
add.data(datum3, estat, T2II, 1, 2, 50)

```

alpha.risk

*False-Alarm Probability for the Three-Sigma R Chart***Description**

Compute the actual in-control false-alarm probability of the conventional three-sigma range chart under the exact distribution of the relative range statistic $W = R/\sigma$.

Usage

```
alpha.risk(n)
```

Arguments

n Numeric vector of integer subgroup sizes, each at least 2.

Details

The conventional limits are

$$D_1\sigma = \max\{0, d_2(n) - 3d_3(n)\}\sigma$$

and

$$D_2\sigma = \{d_2(n) + 3d_3(n)\}\sigma.$$

Because the relative range is not normally distributed for the small subgroup sizes commonly used in practice, the resulting false-alarm probability can be substantially larger than 0.0027.

Value

A numeric vector with the same length and order as `n`. Element i is

$$1 - \{F_W(D_2; n_i) - F_W(D_1; n_i)\},$$

where F_W is evaluated with `stats::ptukey()` using infinite denominator degrees of freedom.

Interpretation

The result is the probability that an in-control subgroup falls below the conventional lower limit or above the conventional upper limit. It is not the nominal value requested by an exact probability-limit design.

Errors

An error is raised when any subgroup size is smaller than 2. The underlying `d2()` and `d3()` calculations also require finite admissible subgroup sizes.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Barbosa, E. P., Gneri, M. A., and Meneguetti, A. (2013). Range control charts revisited: Simpler Tippet-like formulae, its practical implementation, and the study of false alarm. *Communications in Statistics - Simulation and Computation*, 42(2), 247–262. doi:10.1080/03610918.2011.639967.

See Also

[cchart.R](#), [d2](#), [d3](#), [table.qtukey](#)

Examples

```
alpha.risk(5)
alpha.risk(c(3, 5, 10, 15))
```

binomdata

Binomial Process Data for p-Charts

Description

25 subgroups with sample sizes (n_i), number of nonconforming units (D_i), and proportion nonconforming (p_i). This data set is used for p -chart examples.

Format

A data frame with 25 observations on the following 4 variables.

i Index.

ni The sample Size.

Di Number of non-conforming units per sample.

pi Proportion of non-conforming units per sample.

Source

Montgomery, D. C. (2009). *Introduction to Statistical Quality Control*, 6th ed. Wiley.

Examples

```
data(binomdata)
```

c4

Bias Correction Constant for Sample Standard Deviation

Description

Compute the constant c_4 , which corrects the bias of the sample standard deviation S as an estimator of the process standard deviation σ under normality.

Usage

```
c4(n)
```

Arguments

n Integer sample size, at least 2. Can be a vector.

Details

The constant is given by

$$c_4 = \sqrt{\frac{2}{n-1} \frac{\Gamma(n/2)}{\Gamma\{(n-1)/2\}}},$$

so that $E[S] = c_4 \sigma$. Multiplying S by $1/c_4$ produces an unbiased estimator of σ .

Value

Numeric vector of c_4 values, the same length as **n**.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Montgomery, D. C. (2009). *Introduction to Statistical Quality Control*, 6th ed. Wiley.

See Also

[d2](#), [d3](#), [table.const](#)

Examples

```
c4(5)
c4(2:15)
```

cchart.DSnp

Double-Sampling np Control Chart

Description

Build and optionally plot a double-sampling np (DS-np) control chart for monitoring the nonconforming proportion in high-quality processes.

Usage

```
cchart.DSnp(
  x1,
  n1,
  n2,
  p0,
  x2 = NULL,
  w1 = NULL,
  ucl1 = NULL,
  ucl2 = NULL,
  limits = NULL,
  alpha = 0.0027,
  p1 = NULL,
  plot = TRUE,
  curtailed = FALSE,
  ...
)
```

Arguments

x1	Integer vector of nonconforming counts from the first sample.
n1	First-stage sample size (positive integer).
n2	Second-stage sample size (positive integer).
p0	In-control nonconforming proportion.

<code>x2</code>	Optional integer vector of nonconforming counts from the second sample. Must have the same length as <code>x1</code> when provided. Use <code>NA</code> for observations where no second sample was taken. Required for observations where <code>x1</code> falls in the intermediate (warning) zone.
<code>wl</code>	Fractional warning limit. Must be less than <code>uc11</code> .
<code>uc11</code>	Fractional upper control limit for the first stage. Must be greater than <code>wl</code> .
<code>uc12</code>	Fractional upper control limit for the combined samples.
<code>limits</code>	Optional object returned by <code>dsnplimits()</code> . When provided, <code>wl</code> , <code>uc11</code> , and <code>uc12</code> are taken from <code>limits\$best</code> .
<code>alpha</code>	Maximum desired false alarm probability at p_0 . Used only when limits need to be computed via <code>dsnplimits()</code> .
<code>p1</code>	Optional out-of-control proportion. When provided, performance metrics at p_1 are included in the returned object.
<code>plot</code>	Logical. If <code>TRUE</code> (default), draws the control chart. If <code>FALSE</code> , only returns the result object.
<code>curtailed</code>	Logical. If <code>TRUE</code> , use curtailed (truncated) inspection ASS in the performance summary. Default <code>FALSE</code> .
<code>...</code>	Additional arguments passed to <code>plot()</code> .

Details

The DS-np chart uses two sampling stages. At the first stage, a sample of size n_1 is inspected. If the count x_1 is at or below the warning limit, the process is accepted. If x_1 exceeds the first upper control limit, the process signals out-of-control immediately. Otherwise, a second sample of size n_2 is inspected and the combined count $x_1 + x_2$ is compared to the second upper control limit.

Limits can be supplied manually via `wl`, `uc11`, and `uc12`, obtained from a pre-computed `dsnplimits()` object via the `limits` argument, or computed automatically inside the function when neither is provided.

The fractional limits are converted to integer thresholds using the convention from the numerical core functions:

- `wl_accept = floor(wl)`: accept at first stage if $x_1 \leq \text{wl_accept}$.
- `uc11_reject = floor(uc11) + 1`: signal at first stage if $x_1 \geq \text{uc11_reject}$.
- `uc12_accept = floor(uc12)`: accept at second stage if $x_1 + x_2 \leq \text{uc12_accept}$.

Value

An object of class "cchart.DSnp", which is a list with the following elements:

call The matched call.

data A data.frame with columns `index`, `x1`, `x2`, `total`, `stage`, and `signal`.

limits A list with the fractional and integer thresholds: `wl`, `uc11`, `uc12`, `wl_accept`, `uc11_reject`, `uc12_accept`.

parameters A list with n_1 , n_2 , p_0 , α , p_1 , and `curtailed`.

performance A list with in-control performance metrics (`ar10`, `ass0`, `p_signal0`), and optionally out-of-control metrics (`ar11`, `ass1`, `p_signal1`) when p_1 is provided.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Joekes, S., Smrekar, M. and Barbosa, E. (2015). Extending a double sampling control chart for non-conforming proportion in high quality processes to the case of small samples. *Statistical Methodology*.

See Also

[dsnp_limits](#), [dsnp_prob_accept](#), [dsnp_ar1](#), [dsnp_ass](#)

Examples

```
# Small example with manual limits
x1 <- c(0, 1, 2, 3, 1, 0, 2, 4, 1, 0)
x2 <- c(NA, NA, 2, NA, NA, NA, 3, NA, NA, NA)
res <- cchart.DSnp(x1, n1 = 10, n2 = 20, p0 = 0.05,
                  x2 = x2, w1 = 1.5, uc11 = 2.5, uc12 = 4.5,
                  plot = FALSE)

res$limits
res$performance
```

cchart.GV

Generalized Variance Control Chart

Description

Construct and optionally plot a Shewhart-type chart for multivariate process variability based on the generalized variance statistic $|S|$, the determinant of each subgroup sample covariance matrix.

Usage

```
cchart.GV(
  x1,
  size = NULL,
  x2 = NULL,
  Sigma = NULL,
  alpha = 0.0027,
  type = c("normal", "cf", "exact", "simulation"),
  side = c("upper", "two.sided"),
  cf_order = 1,
  nsim = 1e+05,
  seed = NULL,
  plot = TRUE,
  ...
)
```

```
)

## S3 method for class 'cchart.GV'
plot(x, ...)
```

Arguments

x1	Phase I subgroup data in any format accepted by <code>gv_stat()</code> : a list of equal-sized numeric matrices, a subgroup-by-observation-by-variable array, or a numeric matrix containing consecutive subgroups.
size	Positive integer subgroup size when x1 or x2 is a stacked matrix. It is ignored for list and array input.
x2	Optional Phase II subgroup data in a format accepted by <code>gv_stat()</code> . Phase II subgroups must have the same number of observations and variables as the Phase I subgroups.
Sigma	Optional finite symmetric positive-definite $p \times p$ in-control covariance matrix. If supplied, its determinant defines the chart limits. If omitted, the function averages the Phase I covariance matrices and estimates

$$|\Sigma| = |\bar{S}|/b_3,$$

where b_3 is the finite-Phase-I determinant bias correction based on the number of Phase I subgroups, subgroup size, and dimension.

alpha	Nominal probability of a false alarm per subgroup, strictly between 0 and 1. For <code>side = "upper"</code> , all probability is placed in the upper tail. For <code>side = "two.sided"</code> , $\alpha / 2$ is placed in each tail.
type	Character string selecting the limit calculation: "normal" Moment-matched Gaussian quantiles using the exact mean and variance of $ S $. "cf" Cornish-Fisher corrected Gaussian quantiles using skewness, and optionally excess kurtosis. "exact" Closed-form chi-square quantiles for $p = 2$, or the published upper-limit table for $p = 3, n = 4, \dots, 15$, and alpha equal to 0.0020 or 0.0027. "simulation" Monte Carlo quantiles from the Bartlett product-of-chi-squares representation. Partial matching is not used.
side	Either "upper" or "two.sided". Since generalized variance is nonnegative, the lower limit for an upper chart is zero. Any negative approximated lower limit for a two-sided chart is also truncated to zero.
cf_order	Integer 1 or 2. Order 1 uses the skewness correction. Order 2 additionally uses excess kurtosis and the squared-skewness term. It is used only when <code>type = "cf"</code> .
nsim	Integer number of Monte Carlo draws, at least 1000. It is used by <code>type = "simulation"</code> and stored in the returned object for all methods.

seed	NULL or a finite numeric scalar used as the Monte Carlo seed. A supplied seed makes simulated limits reproducible and the caller's existing <code>.Random.seed</code> is restored on exit.
plot	Logical scalar. If TRUE, draw the chart before returning the result. If FALSE, construct the object without plotting.
...	Additional graphical arguments passed to <code>plot.cchart.GV()</code> and then to <code>graphics::plot()</code> .
x	An object of class "cchart.GV".

Details

Phase I subgroups define the chart design and, unless `Sigma` is supplied, estimate the in-control generalized variance. Optional Phase II subgroups are then evaluated using the same subgroup size, process dimension, and fixed control limits.

Each subgroup must contain finite observations on at least two variables, and the subgroup size must exceed the process dimension. When `Sigma = NULL`, the average Phase I covariance matrix must itself be positive definite. The control-line center is the exact in-control mean of $|S|$, not generally $|\Sigma|$.

`plot.cchart.GV()` displays subgroup generalized variances, the center line, lower and upper limits, a vertical separator before Phase II, and solid points at signaled subgroups. The plot method returns its input invisibly.

Value

An object of class "cchart.GV", a list with components:

`statistics` Generalized variances for all Phase I followed by all Phase II subgroups.

`phase1, phase2` Integer indices identifying the two phases in `statistics`.

`limits` The complete list returned by `gv_limits()`.

`out.of.control` Indices for which `statistics < lcl` or `statistics > ucl`. Equality to a limit does not signal.

`Sbar` The average Phase I covariance matrix when `Sigma` is estimated, otherwise NULL.

`Sigma` The supplied in-control covariance matrix, or NULL when the determinant is estimated from Phase I.

`b3` The determinant bias-correction factor when Phase I estimation is used, otherwise `NA_real_`.

`n, p` Subgroup size and process dimension.

`call` The matched function call.

For `plot.cchart.GV()`, `x` invisibly.

Phase I and Phase II convention

Phase I data are used both as historical plotted points and, when `Sigma` is absent, to estimate the in-control determinant. Phase II data do not alter the estimate or the limits. For a strict prospective chart with known parameters, supply `Sigma`.

Simulated limits

When `type = "simulation"`, `nsim` and `seed` are passed to `gv_limits()`. A supplied seed makes the result reproducible and the caller's existing random-number state is restored.

Errors

Errors are raised for invalid subgroup representations, unequal subgroup dimensions, non-finite observations, $n \leq p$, an invalid or non-positive-definite `Sigma`, a singular Phase I average covariance matrix, and any unsupported limit configuration documented in `gv_limits()`.

References

Barbosa, E. P., Gneri, M. A., and Meneguetti, A. *Improving Shewhart-type Generalized Variance Control Charts for Multivariate Process Variability Monitoring using Cornish-Fisher Quantile Correction, Meijer-G Function and Other Tools*. Research report.

Alt, F. B. (1984). Multivariate quality control. In Johnson, N. L. and Kotz, S. (eds.), *Encyclopedia of Statistical Sciences*, Vol. 6, 110–122. Wiley.

See Also

[gv_stat](#), [gv_limits](#), [gv_alpha_risk](#), [print.cchart.GV](#)

Examples

```
set.seed(123)
phase1 <- array(rnorm(6 * 8 * 2), dim = c(6, 8, 2))

# Known in-control covariance, exact dimension-two limits.
chart <- cchart.GV(
  phase1,
  Sigma = diag(2),
  type = "exact",
  plot = FALSE
)
chart$limits[c("lcl", "center", "ucl")]
summary(chart)

# Estimate |Sigma| from Phase I and evaluate two Phase II subgroups.
phase2 <- array(rnorm(2 * 8 * 2), dim = c(2, 8, 2))
estimated <- cchart.GV(
  phase1,
  x2 = phase2,
  type = "cf",
  plot = FALSE
)
estimated$phase2
```

cchart.p

*p-chart***Description**

Build normal, Cornish-Fisher corrected, or standardized p charts.

Usage

```
cchart.p(
  x1 = NULL,
  n1 = NULL,
  type = "norm",
  p1 = NULL,
  x2 = NULL,
  n2 = NULL,
  phat = NULL,
  p2 = NULL,
  alpha = ALPHA
)
```

Arguments

x1	Phase I nonconforming counts.
n1	Phase I sample size or vector of sample sizes.
type	Chart type. Accepted values are "normal", "cf1", "cf2", and "standardized". The legacy aliases "norm", "CF", and "std" remain supported; "CF" maps to "cf1".
p1	Phase I subgroup proportions. Used instead of x1.
x2	Phase II nonconforming counts.
n2	Phase II sample size or vector of sample sizes.
phat	Known or previously estimated in-control proportion.
p2	Phase II subgroup proportions. Used instead of x2.
alpha	Nominal two-sided false alarm probability. Defaults to 0.0027.

Details

For a Phase I chart, n1 and exactly one of x1 or p1 must be supplied. For a Phase II chart, n2 and exactly one of x2 or p2 must be supplied, together with Phase I information or a known phat.

When sample sizes vary, the process proportion is estimated by the pooled binomial estimator, $\text{sum}(x_i)/\text{sum}(n_i)$, rather than by the unweighted mean of subgroup proportions. The plotting wrapper uses two-sided limits; use `pchart_limits()` directly for one-sided upper limits.

Value

Invisibly returns the qcc object used to draw the chart.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Montgomery, D. C. (2008). *Introduction to Statistical Quality Control*. Wiley.

Joekes, S. and Barbosa, E. P. (2013). An improved attribute control chart for monitoring non-conforming proportion in high quality processes. *Control Engineering Practice*, 21, 407–412. [doi:10.1016/j.conengprac.2012.12.005](https://doi.org/10.1016/j.conengprac.2012.12.005).

Examples

```
data(binomdata)
cchart.p(x1 = binomdata$Di[1:12], n1 = binomdata$ni[1:12])
cchart.p(x1 = binomdata$Di[1:12], n1 = binomdata$ni[1:12],
         type = "cf2", x2 = binomdata$Di[13:25],
         n2 = binomdata$ni[13:25])
cchart.p(type = "standardized", p2 = binomdata$Di[13:25] /
         binomdata$ni[13:25], n2 = binomdata$ni[13:25],
         phat = 0.1115833)
```

cchart.R

Range Control Chart

Description

Build a control chart for subgroup ranges using either the conventional three-sigma approximation or exact probability limits from the distribution of the relative range $W = R/\sigma$.

Usage

```
cchart.R(x, n, type = c("norm", "tukey"), y = NULL)
```

Arguments

x	Phase II subgroup data accepted by <code>qcc::qcc()</code> for an "R" chart. Rows represent subgroups and columns observations within subgroups.
n	Integer subgroup size, at least 2. It is used to evaluate the actual false-alarm probability of the conventional chart and to obtain Tukey relative-range quantiles for the exact chart.
type	Either "norm" for the conventional three-sigma chart or "tukey" for exact equal-tail probability limits.

`y` Phase I subgroup data used to estimate σ through `qcc::sd.R()` when `type = "tukey"`. It must be supplied for that method and have the same subgroup structure as `x`.

Details

The numerical limits are computed by `r_shewhart_limits` (for `type = "norm"`) or `r_exact_limits` (for `type = "tukey"`), keeping the plotting wrapper separate from the pure limit calculations.

For `type = "norm"`, limits are delegated to the standard `qcc` range-chart implementation. For `type = "tukey"`, the lower and upper limits are

$$\hat{\sigma} F_W^{-1}(0.00135; n)$$

and

$$\hat{\sigma} F_W^{-1}(0.99865; n),$$

where $\hat{\sigma}$ is estimated from `y` and the quantiles are obtained with `stats::qtukey()`.

Value

Invisibly, the "qcc" object returned by `qcc::qcc()`. The function also draws the chart. For `type = "norm"`, a message is added below the plot showing the actual false-alarm probability returned by `alpha.risk(n)`.

Phase convention

The exact chart treats `y` as Phase I reference data and `x` as the plotted monitoring data. The conventional chart uses the standard estimation behavior of `qcc::qcc()` on `x`.

Errors

An error is raised for an unsupported `type`, for `n < 2`, or when `y` is omitted for the exact chart. Additional data validation is performed by `qcc::qcc()` and `qcc::sd.R()`.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Barbosa, E. P., Gneri, M. A., and Meneguetti, A. (2013). Range control charts revisited: Simpler Tippett-like formulae, its practical implementation, and the study of false alarm. *Communications in Statistics - Simulation and Computation*, 42(2), 247–262. doi:10.1080/03610918.2011.639967.

See Also

`r_shewhart_limits`, `r_exact_limits`, `alpha.risk`, `cchart.S`, `table.qtukey`

Examples

```
data(pistonrings)
conventional <- cchart.R(pistonrings[1:25, ], 5)
exact <- cchart.R(
  pistonrings[26:40, ], 5,
  type = "tukey",
  y = pistonrings[1:25, ]
)
```

cchart.S

Standard-Deviation Control Chart

Description

Build a control chart for subgroup standard deviations using either the conventional Shewhart limits used historically by qcc or exact probability limits derived from the chi-square distribution of the sample variance. Numerical limits are delegated to [s_shewhart_limits](#) and [s_exact_limits](#).

Usage

```
cchart.S(x, type = c("n", "e"), m = NULL)
```

Arguments

x	Subgroup data accepted by <code>qcc::qcc()</code> for an "S" chart. Rows represent subgroups and columns observations within subgroups.
type	Either "n" for the normalized qcc-compatible limits or "e" for exact equal-tail probability limits.
m	Integer subgroup size, at least 2. It is required when <code>type = "e"</code> . If omitted, a warning is issued and the normalized chart is drawn instead. This argument is retained unchanged for compatibility with historical positional calls.

Details

For exact limits, $(m - 1)S^2/\sigma^2$ follows a chi-square distribution with $m - 1$ degrees of freedom under normality. The historical wrapper estimates σ from `x` using `qcc::sd.S()` and supplies that estimate to `s_exact_limits()`; these are therefore plug-in probability limits.

For `type = "n"`, the legacy qcc convention is deliberately preserved: the center is the weighted mean of the observed subgroup standard deviations, while σ is estimated separately by `qcc::sd.S()`. The resulting standard error is $\hat{\sigma}\sqrt{1 - c_4(n)^2}$. This differs in finite Phase I samples from replacing the observed center by the theoretical $c_4(n)\hat{\sigma}$; `cchart.S()` does not make that replacement.

Value

Invisibly, the "qcc" object returned by `qcc::qcc()`. The function also draws the chart.

Phase convention

The existing API has no separate Phase I argument. Both the chart center and σ estimate are obtained from x ; exact limits additionally use the supplied historical design size m . Thus both normalized and exact limits are plug-in limits when used through this wrapper.

Errors and warnings

An error is raised for an unsupported type or an invalid supplied m . If exact limits are requested without m , the function warns and falls back to the normalized chart. Additional data validation is performed by `qcc::qcc()`, `qcc::stats.S()`, and `qcc::sd.S()`.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Montgomery, D. C. (2009). *Introduction to Statistical Quality Control*, 6th ed. Wiley.

See Also

[s_shewhart_limits](#), [s_exact_limits](#), [cchart.R](#), [c4](#)

Examples

```
data(softdrink)
normalized <- cchart.S(softdrink, type = "n")
exact <- cchart.S(softdrink, type = "e", m = 10)
```

cchart.T2.1

Phase I Hotelling T^2 Control Chart

Description

Build a Phase I T^2 control chart. For $n = 1$: $UCL = ((m - 1)^2 / m) \times qbeta(1 - \alpha, p/2, (m - p - 1)/2)$. For $n > 1$: $UCL = (p(m - 1)(n - 1)) / (mn - m - p + 1) \times qf(1 - \alpha, p, mn - m - p + 1)$. The control limits are based on the beta and F distributions.

Usage

```
cchart.T2.1(T2, m, n, p)
```

Arguments

T2	Numeric vector of T^2 statistics from T2.1 .
m	Number of subgroups in Phase I.
n	Subgroup size. Use $n = 1$ for individual observations, $n > 1$ for subgroups.
p	Number of variables (dimension).

Value

Draws the chart. Returns nothing.

Phase convention

Phase I retrospective analysis. All points are plotted simultaneously.

Errors

Stop if $n < 1$.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Montgomery, D.C., (2009). "Introduction to Statistical Quality Control". Chapter 11. Wiley.

See Also

[T2.1](#), [cchart.T2.2](#), [robustez](#)

Examples

```
mu <- c(5.682, 88.22)
Sigma <- miscTools::symMatrix(c(3.770, -5.495, 13.53), 2)
datum <- data.1(20, 10, mu, Sigma)
estat <- stats(datum, 20, 10, 2)
T2 <- T2.1(estat, 20, 10)
# estat is a list with the auxiliary statistics. T2 is a matrix with the values of the T2 statistic.
cchart.T2.1(T2, 20, 10, 2)
```

cchart.T2.2

Phase II Hotelling T² Control Chart

Description

Build or update a Phase II T² chart. Can optionally display Phase I reference data. Limits: For $n = 1$: $UCL = (p(m+1)(m-1))/(m^2 - mp) \times qf(1 - \alpha, p, m - p)$. For $n > 1$: $UCL = (p(m+1)(n-1))/(mn - m - p + 1) \times qf(1 - \alpha, p, mn - m - p + 1)$.

Usage

```
cchart.T2.2(T2II, m, n, j, t, p, datum = NULL, stats = NULL, T2 = NULL)
```

Arguments

T2II	Vector with Phase II T^2 values (from T2.2).
m	Number of Phase I subgroups.
n	Subgroup size.
j	Index of the current Phase II sample (1-based).
t	Maximum x-axis limit (number of Phase II samples to show).
p	Number of variables.
datum	Optional. Phase I data array to display reference points.
stats	Optional. Phase I statistics to compute T^2 if not provided.
T2	Optional. Pre-computed Phase I T^2 values.

Value

Draws the chart. Returns nothing.

Phase convention

Phase II monitoring. The vertical dashed line separates Phase I (left) from Phase II (right).

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Montgomery, D.C., (2009). "Introduction to Statistical Quality Control". Chapter 11. Wiley.

See Also

[T2.2](#), [cchart.T2.1](#), [robustez](#)

Examples

```
mu <- c(5.682, 88.22)
Sigma <- miscTools::symMatrix(c(3.770, -5.495, 13.53), 2)
datum <- data.1(20, 10, mu, Sigma)
estat <- stats(datum, 20, 10, 2)
datum2 <- data.2(estat, 10, p = 2)
T2II <- T2.2(datum2, estat, 10)
# For the first sample j = 1. T2II is a vector with the value of the first T2 statistic.
cchart.T2.2(T2II, 20, 10, 1, 25, 2)
# Same of the above, but now showing the phase I data set.
cchart.T2.2(T2II, 20, 10, 1, 25, 2, datum = datum)

#Example with individual observations
datum <- data.1(50, 1, mu, Sigma)
estat <- stats(datum, 50, 1, 2)
datum2 <- data.2(estat, 1, p = 2)
```

```

T2II <- T2.2(datum2, estat, 1)
# For the first sample j = 1. T2II is a vector with the value of the first T2 statistic.
cchart.T2.2(T2II, 50, 1, 1, 25, 2)
# Same of the above, but now showing the phase I data set.
cchart.T2.2(T2II, 50, 1, 1, 25, 2, datum = datum)

```

cchart.trV

Trace-Statistic Control Chart

Description

Construct and optionally plot a Shewhart-type upper control chart for multivariate process variability using the standardized trace statistic $tr(V)$.

Usage

```

cchart.trV(
  x,
  size = NULL,
  Sigma0 = NULL,
  alpha = 0.0027,
  type = c("chisq", "simulation"),
  nsim = 1e+05,
  seed = NULL,
  newdata = NULL,
  plot = TRUE,
  ...
)

## S3 method for class 'cchart.trV'
plot(x, ...)

## S3 method for class 'cchart.trV'
print(x, digits = max(3L, getOption("digits") - 3L), ...)

## S3 method for class 'cchart.trV'
summary(object, ...)

## S3 method for class 'summary.cchart.trV'
print(x, digits = max(3L, getOption("digits") - 3L), ...)

```

Arguments

x An object of class "summary.cchart.trV".

size Positive integer subgroup size when x or newdata is a stacked matrix.

<code>Sigma0</code>	Optional finite symmetric positive-definite in-control covariance matrix. If omitted, the average Phase I covariance matrix is used as a plug-in estimate.
<code>alpha</code>	Nominal upper-tail false-alarm probability, strictly between 0 and 1.
<code>type</code>	Character string selecting the calculation: "chisq" Exact chi-square quantile with $p(n - 1)$ degrees of freedom. "simulation" Monte Carlo quantile from the same chi-square distribution. This is mainly a diagnostic and teaching option.
<code>nsim</code>	Integer number of Monte Carlo draws, at least 1000. Used only when <code>type = "simulation"</code> .
<code>seed</code>	NULL or a finite numeric scalar used as the Monte Carlo seed. A supplied seed makes simulated limits reproducible and restores the caller's existing random-number state on exit.
<code>newdata</code>	Optional Phase II subgroup data in any format accepted by <code>trv_stat</code> . Phase II subgroups must have the same subgroup size and dimension as Phase I.
<code>plot</code>	Logical scalar. If TRUE, draw the chart before returning the result.
<code>...</code>	Currently unused.
<code>digits</code>	Number of significant digits used for numerical output.
<code>object</code>	An object of class <code>"cchart.trV"</code> .

Details

The `tr(V)` chart complements `cchart.GV`. The generalized variance chart monitors determinant changes, while the trace chart monitors the sum of standardized covariance eigenvalues. When `Sigma0 = NULL`, limits are plug-in limits conditional on the Phase I estimate and do not include additional Phase I estimation uncertainty.

Value

An object of class `"cchart.trV"`, a list with components:

`statistics` Trace statistics for all Phase I followed by all Phase II subgroups.

`phase1, phase2` Integer indices identifying the two phases in `statistics`.

`limits` The complete list returned by `trv_limits()`.

`out.of.control` Indices for which `statistics > ucl`. Equality to the limit does not signal.

`Sigma0` The supplied or estimated in-control covariance matrix.

`Sbar` The average Phase I covariance matrix when `Sigma0` is estimated, otherwise NULL.

`n, p` Subgroup size and process dimension.

`call` The matched function call.

For `plot.cchart.trV()`, `x` invisibly.

`x`, invisibly.

An object of class `"summary.cchart.trV"`.

`x`, invisibly.

Examples

```
set.seed(123)
phase1 <- array(rnorm(6 * 8 * 2), dim = c(6, 8, 2))
chart <- cchart.trV(phase1, Sigma0 = diag(2), plot = FALSE)
chart$limits[c("center", "ucl")]
```

cchart.u

*u-chart***Description**

Build normal, Cornish-Fisher corrected, or standardized u charts.

Usage

```
cchart.u(
  x1 = NULL,
  n1 = NULL,
  type = "norm",
  u1 = NULL,
  x2 = NULL,
  n2 = NULL,
  lambda = NULL,
  u2 = NULL,
  alpha = ALPHA
)
```

Arguments

x1	Phase I defect counts.
n1	Phase I inspection size or vector of sizes.
type	Chart type: "normal", "cf1", "cf2", or "standardized". Legacy aliases "norm", "CF", and "std" remain supported; "CF" maps to "cf2" because the historical IQCC formula contains both Cornish-Fisher adjustments.
u1	Phase I rates, used instead of x1.
x2	Phase II defect counts.
n2	Phase II inspection size or vector of sizes.
lambda	Known or estimated in-control defect rate.
u2	Phase II rates, used instead of x2.
alpha	Nominal two-sided false-alarm probability.

Details

For Phase I, supply n1 and exactly one of x1 or u1. For Phase II, supply n2 and exactly one of x2 or u2, together with Phase I information or a known lambda.

When inspection sizes vary, the in-control rate is estimated by the pooled Poisson estimator $\text{sum}(x_i)/\text{sum}(n_i)$.

Value

Invisibly returns the qcc object used to draw the chart.

Examples

```
data(moonroof)
cchart.u(x1 = moonroof$yi[1:17], n1 = moonroof$ni[1:17])
cchart.u(x1 = moonroof$yi[1:17], n1 = moonroof$ni[1:17],
         type = "CF", x2 = moonroof$yi[18:34], n2 = moonroof$ni[18:34])
cchart.u(type = "std", u2 = moonroof$ui[18:34],
         n2 = moonroof$ni[18:34], lambda = 1.4)
```

cchart.Xbar

X-bar Control Chart for Phase I and Phase II

Description

Builds and displays an X-bar control chart for Phase I (retrospective analysis), Phase II (monitoring of future production), or both phases in a single call. When Phase I data is supplied without Phase II data, the chart uses estimated control limits. If Phase II data is supplied without Phase I data, the chart requires x2bars and sigma as reference values.

Usage

```
cchart.Xbar(
  x1 = NULL,
  n1 = NULL,
  x2 = NULL,
  n2 = NULL,
  x2bars = NULL,
  sigma = NULL
)
```

Arguments

x1	Phase I data. A matrix or data frame where each row is a subgroup.
n1	Phase I subgroup size(s). A single integer (equal sizes) or a vector.
x2	Phase II data. Same structure as x1.
n2	Phase II subgroup size(s).
x2bars	Mean of subgroup means from Phase I (center line). Can be NULL and computed from x1.
sigma	Standard deviation from Phase I (for control limits). Can be NULL and computed from x1.

Value

Draws the X-bar control chart using [qcc](#). For Phase I only, invisibly returns a list with components x2bar (mean of means) and sigma (standard deviation).

Phase convention

Phase I when only Phase I data is supplied; limits are estimated from the data. Phase II when Phase II data is supplied (with or without Phase I reference data).

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Montgomery, D.C., (2009). "Introduction to Statistical Quality Control". Chapter 6. Wiley.

See Also

[cchart.Xbar1](#), [cchart.Xbar2](#), [cchart.Xbar_R](#)

Examples

```
data(pistonrings)
cchart.Xbar(x1 = pistonrings[1:25, ], n1 = 5)
cchart.Xbar(x1 = pistonrings[1:25, ], n1 = 5, x2 = pistonrings[26:40, ], n2 = 5)
```

cchart.Xbar_R	<i>X-bar and R Control Charts</i>
---------------	-----------------------------------

Description

Draws the X-bar control chart and the R (range) control chart side by side in the same graphics window, using estimated Phase I control limits. Both charts share the same subgroup data.

Usage

```
cchart.Xbar_R(x, sizes)
```

Arguments

- x Phase I data. Matrix or data frame with subgroups in rows.
- sizes Subgroup size(s). A single integer (equal sizes) or a vector.

Value

Draws two control charts side by side. Returns nothing.

Phase convention

Phase I --- control limits estimated from the data for both charts.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Montgomery, D.C., (2009). "Introduction to Statistical Quality Control". Chapter 6. Wiley.

See Also

[cchart.Xbar1](#), [cchart.Xbar2](#), [cchart.Xbar](#)

Examples

```
data(pistonrings)
attach(pistonrings)
cchart.Xbar_R(pistonrings[1:25, ], 5)
```

cchart.Xbar1

Phase I X-bar Shewhart Control Chart

Description

Builds the X-bar control chart for Phase I retrospective analysis. The control limits are estimated from the supplied data. Even if the data is not normal, the X-bar statistic is approximately normal by the central limit theorem when subgroup sizes are sufficiently large.

Usage

```
cchart.Xbar1(x, sizes)
```

Arguments

x	Phase I data. Matrix or data frame with subgroups in rows.
sizes	Subgroup size(s). A single integer (equal sizes) or a vector.

Value

A list with components x2bar (mean of subgroup means) and sigma (standard deviation), returned invisibly. The control chart is drawn as a side effect.

Phase convention

Phase I --- control limits estimated from the data.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Montgomery, D.C., (2009). "Introduction to Statistical Quality Control". Chapter 6. Wiley.

See Also

[cchart.Xbar2](#), [cchart.Xbar](#), [cchart.Xbar_R](#)

Examples

```
data(pistonrings)
cchart.Xbar1(pistonrings[1:25, ], 5)
```

cchart.Xbar2

Phase II X-bar Shewhart Control Chart

Description

Builds the X-bar control chart for Phase II monitoring using reference limits obtained from a Phase I analysis. The Phase I estimates of `x2bar` (mean of means) and `sigma` (standard deviation) are required, typically obtained from [cchart.Xbar1](#).

Usage

```
cchart.Xbar2(x, x2bar, sigma, sizes)
```

Arguments

<code>x</code>	Phase II data. Matrix or data frame with subgroups in rows.
<code>x2bar</code>	Mean of subgroup means from Phase I (center line).
<code>sigma</code>	Standard deviation from Phase I (for control limits).
<code>sizes</code>	Phase II subgroup size(s).

Value

Draws the X-bar control chart. Returns nothing.

Phase convention

Phase II --- tests new data against Phase I reference distribution.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Montgomery, D.C., (2009). "Introduction to Statistical Quality Control". Chapter 6. Wiley.

See Also

[cchart.Xbar1](#), [cchart.Xbar](#)

Examples

```
data(pistonrings)
stat <- cchart.Xbar1(pistonrings[1:25, ], 5)
cchart.Xbar2(pistonrings[26:40, ], stat[[1]][1], stat[[1]][2], 5)
```

d_2	<i>Mean of the Relative Range</i>
-------	-----------------------------------

Description

Compute the constant d_2 , the expected value of the relative range $W = R/\sigma$ for a sample of size n drawn from a normal distribution, where R is the sample range and σ is the process standard deviation.

Usage

`d2(n)`

Arguments

`n` Integer sample size, at least 2. Can be a vector.

Details

Under normality, W has the studentized range distribution with infinite denominator degrees of freedom. Its expectation is the integral of the survival function, $E[W] = \int_0^\infty \{1 - F_W(w)\}dw$, which is evaluated numerically using [ptukey](#) with infinite degrees of freedom.

Value

Numeric vector of d_2 values, the same length as `n`.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

- Montgomery, D. C. (2009). *Introduction to Statistical Quality Control*, 6th ed. Wiley.
- Barbosa, E. P., Gneri, M. A. and Meneguetti, A. (2013). Range control charts revisited: Simpler Tippett-like formulae, its practical implementation, and the study of false alarm. *Communications in Statistics - Simulation and Computation*, 42(2), 247–262. doi:10.1080/03610918.2011.639967.

See Also

[d3](#), [c4](#), [table.const](#)

Examples

```
d2(5)
d2(2:10)
```

d3

Standard Deviation of the Relative Range

Description

Compute the constant d_3 , the standard deviation of the relative range $W = R/\sigma$ for a sample of size n drawn from a normal distribution, where R is the sample range and σ is the process standard deviation.

Usage

```
d3(n)
```

Arguments

n Integer sample size, at least 2. Can be a vector.

Details

Under normality, W has the studentized range distribution with infinite denominator degrees of freedom. Using the identity $E[W^2] = 2 \int_0^\infty w\{1 - F_W(w)\}dw$, the standard deviation is obtained as $d_3 = \sqrt{E[W^2] - d_2^2}$, where d_2 is computed by [d2](#). The integrals are evaluated numerically using [ptukey](#) with infinite degrees of freedom.

Value

Numeric vector of d_3 values, the same length as **n**.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

- Montgomery, D. C. (2009). *Introduction to Statistical Quality Control*, 6th ed. Wiley.
- Barbosa, E. P., Gneri, M. A. and Meneguetti, A. (2013). Range control charts revisited: Simpler Tippett-like formulae, its practical implementation, and the study of false alarm. *Communications in Statistics - Simulation and Computation*, 42(2), 247–262. doi:10.1080/03610918.2011.639967.

See Also

d2, c4

Examples

```
d3(7)
d3(2:10)
```

data.1	<i>Simulate Phase I Multivariate Normal Data</i>
--------	--

Description

Generate m samples from a p -variate normal distribution for Phase I Hotelling T^2 chart development. For $n = 1$: returns an $m \times p$ matrix. For $n > 1$: returns an $n \times p \times m$ array.

Usage

```
data.1(m, n, mu, Sigma)
```

Arguments

m	Number of samples.
n	Sample size per subgroup. Use $n = 1$ for individual observations, $n > 1$ for subgroups.
mu	Mean vector (length p).
Sigma	$p \times p$ covariance matrix.

Value

For $n = 1$: an $m \times p$ matrix. For $n > 1$: an $n \times p \times m$ array.

Simulation

Uses MASS::mvrnorm(). Set the seed externally for reproducibility.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

See Also[data.2](#), [T2.1](#)**Examples**

```
mu <- c(5.682, 88.22)
Sigma <- miscTools::symMatrix(c(3.770, -5.495, 13.53), 2)
#Simulated data with individual observations
datum <- data.1(50, 1, mu, Sigma)
#Simulated data with sub-group observations
datum <- data.1(20, 10, mu, Sigma)
```

data.2

*Simulate Phase II Multivariate Normal Data***Description**

Generate a single new observation from $N(\mu_0 + \delta, \Sigma)$ using Phase I reference parameters for Phase II monitoring.

Usage

```
data.2(estat, n, delta = 0, p)
```

Arguments

estat	List from stats (Phase I estimates).
n	Subgroup size. Use $n = 1$ for individual observations, $n > 1$ for subgroups.
delta	Optional shift added to mean vector for out-of-control simulation.
p	Dimension.

Details

To use this function it is necessary to have the information about Phase I given by the functions [data.1](#) and [stats](#).

Value

For $n = 1$: a numeric vector. For $n > 1$: an $n \times p$ matrix.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

See Also[data.1](#), [T2.2](#)

Examples

```
mu <- c(5.682, 88.22)
Sigma <- miscTools::symMatrix(c(3.770, -5.495, 13.53), 2)
datum <- data.1(20, 10, mu, Sigma)
# estat is the list with the values of the auxiliary statistics.
estat <- stats(datum, 20, 10, 2)
datum2 <- data.2(estat, 10, p = 2)
```

dsnp_arl

Double-Sampling np Chart: Average Run Length

Description

Compute the average run length for the double-sampling np chart as the reciprocal of the signal probability at each supplied nonconforming proportion.

Usage

```
dsnp_arl(p, n1, n2, w1, ucl1, ucl2)
```

Arguments

p	Nonconforming proportion to evaluate, a finite numeric scalar or vector in $[0, 1]$.
n1	First-stage sample size, a positive integer.
n2	Second-stage sample size, a positive integer.
w1	Finite fractional warning limit.
ucl1	Finite fractional first-stage upper control limit greater than w1.
ucl2	Finite fractional combined upper control limit greater than w1.

Details

For signal probability $1 - P_T(p)$, the ARL is $1/(1 - P_T(p))$. It is infinite when the signal probability is zero.

Value

A list with arl, the average run length; pt, total acceptance probability; p_signal, signal probability; and the validated chart parameters n1, n2, w1, ucl1, and ucl2.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Joekes, S., Smrekar, M. and Barbosa, E. P. (2015). Extending a double sampling control chart for non-conforming proportion in high quality processes to the case of small samples. *Statistical Methodology*, 23, 35–49.

See Also

[dsnp_prob_accept](#), [dsnp_ass](#)

Examples

```
ar10 <- dsnp_ar1(0.005, 34, 162, 1.5, 2.5, 4.5)$ar1
ar11 <- dsnp_ar1(0.0075, 34, 162, 1.5, 2.5, 4.5)$ar1
c(ar10 = ar10, ar11 = ar11)
```

dsnp_ass	<i>Double-Sampling np Chart: Average Sample Size</i>
----------	--

Description

Compute the average sample size for the double-sampling np chart.

Usage

```
dsnp_ass(p, n1, n2, w1, ucl1, ucl2 = NULL, curtailed = FALSE)
```

Arguments

p	Nonconforming proportion to evaluate, a finite numeric scalar or vector in $[0, 1]$.
n1	First-stage sample size, a positive integer.
n2	Second-stage sample size, a positive integer.
w1	Finite fractional warning limit.
ucl1	Finite fractional first-stage upper control limit greater than w1.
ucl2	Finite fractional second-stage upper control limit. Required when curtailed = TRUE.
curtailed	Logical. If FALSE (default), assume complete inspection of every second-stage sample. If TRUE, use curtailed (truncated) inspection within the second sample.

Details

By default (`curtailed = FALSE`), every second-stage sample that is requested is fully inspected. Therefore

$$ASS(p) = n_1 + n_2 P_p(\text{second stage}).$$

When `curtailed = TRUE`, inspection of the second sample stops as soon as the cumulative count of non-conformities exceeds `ucl2`. For each warning-zone first-stage count d_1 , define $r(d_1) = \lfloor ucl_2 \rfloor - d_1 + 1$ as the number of non-conformances needed to reject. The expected number of stage-2 items inspected is

$$E[M(d_1)] = \sum_{j=0}^{n_2-1} P(\text{Bin}(j, p) \leq r(d_1) - 1),$$

with $E[M(d_1)] = 0$ when $r(d_1) \leq 0$. Then

$$ASS_{\text{curtailed}}(p) = n_1 + \sum_{d_1=a+1}^{b-1} P(D_1 = d_1) E[M(d_1)].$$

Curtailed inspection does not change the signal probability or ARL; it only reduces the number of items inspected when the eventual decision is already determined before the full second sample is observed.

Value

A list with:

ass Average sample size (numeric vector).

p_second Probability that the second sample is required.

n1, n2, wl, ucl1, ucl2 Validated chart parameters.

curtailed The convention used.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Joekes, S., Smrekar, M. and Barbosa, E. P. (2015). Extending a double sampling control chart for non-conforming proportion in high quality processes to the case of small samples. *Statistical Methodology*, 23, 35–49. doi:10.1016/j.stamet.2014.09.003.

See Also

[dsnp_prob_accept](#), [dsnp_arl](#)

Examples

```
dsnp_ass(0.005, 34, 162, 1.5, 2.5)$ass

# Curtailed inspection
dsnp_ass(0.005, 34, 162, 1.5, 2.5, ucl2 = 4.5, curtailed = TRUE)$ass
```

dsnp_design

*Double-Sampling np Chart: Complete Design Search***Description**

Perform a joint discrete search over first-stage sample size (n_1), second-stage sample size (n_2), and DS-np control limits to find optimal chart designs. This function performs an exhaustive search over the Cartesian product of n_1 _range and n_2 _range, calling [dsnp_limits](#) for each pair and collecting the best feasible candidates across all pairs.

Usage

```
dsnp_design(
  p0,
  p1,
  n1_range,
  n2_range,
  arl0_min = 200,
  alpha = NULL,
  objective = c("ar1", "ass0", "weighted"),
  weights = c(ar1 = 1, ass0 = 1),
  allow_empty_warning = FALSE,
  max_results = 20,
  progress = FALSE,
  ass0_max = NULL,
  curtailed = FALSE
)

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

Arguments

p_0	In-control nonconforming proportion. Scalar in (0, 1).
p_1	Out-of-control nonconforming proportion. Scalar in (0, 1). Must be greater than p_0 .
n_1 _range	Integer vector of first-stage sample sizes to evaluate. Positive integers, no duplicates, no NAs.

<code>n2_range</code>	Integer vector of second-stage sample sizes to evaluate. Positive integers, no duplicates, no NAs.
<code>ar10_min</code>	Minimum acceptable in-control ARL. Scalar greater than 1, or NULL if <code>alpha</code> is supplied.
<code>alpha</code>	Maximum acceptable false alarm probability. Scalar in (0, 1), or NULL if <code>ar10_min</code> is supplied. When both are provided, candidates must satisfy both constraints.
<code>objective</code>	Optimization objective. One of "ar11" (minimize out-of-control ARL), "ass0" (minimize in-control ASS), or "weighted" (minimize a weighted combination of normalized ARL1 and ASS0).
<code>weights</code>	Named numeric vector with elements <code>ar11</code> and <code>ass0</code> . Both must be non-negative and at least one must be positive. Only used when <code>objective</code> = "weighted".
<code>allow_empty_warning</code>	Logical. If FALSE (default), discard candidates with no integer in the warning zone.
<code>max_results</code>	Maximum number of feasible candidates to return. Positive integer.
<code>progress</code>	Logical. If TRUE, print progress messages during the search.
<code>ass0_max</code>	Optional maximum in-control average sample size. When supplied, it must be a finite positive scalar and candidates must satisfy $ass0 \leq ass0_max$. The ASS calculation follows equation (15) of Joeke et al. (2015); when <code>curtailed</code> = TRUE the expected inspection of the second sample reflects truncated (curtailed) inspection.
<code>curtailed</code>	Logical. If TRUE, use curtailed (truncated) inspection ASS instead of complete second-sample ASS. Default FALSE.
<code>x</code>	An object of class "dsnp_design".
<code>...</code>	Additional arguments (ignored).

Details

The search is discrete and exhaustive within the supplied ranges; it is not a continuous optimization. The cost grows with the number of (n1, n2) pairs and the number of limit candidates evaluated per pair.

When `alpha` is NULL and `ar10_min` is provided, an effective `alpha` of $1 / ar10_min$ is used to guide `dsnp_limits()`, but final feasibility is checked against the explicit $ar10 \geq ar10_min$ condition.

The published design problem minimizes out-of-control ARL subject to $ass0 \leq ass0_max$ and $ar10 \geq ar10_min$. Setting `ass0_max` = NULL omits only the ASS constraint and preserves the behavior of earlier IQCC versions. The argument is appended to the function signature so existing positional calls retain their meaning.

Value

An object of class "dsnp_design" with the following elements:

best The top-ranked feasible candidate (a one-row data.frame).

candidates Up to `max_results` feasible candidates, sorted by the chosen objective.

parameters A list of the input parameters.

search A list with search summary counts:

n_pairs_evaluated Number of (n1, n2) pairs evaluated.

n_candidates_evaluated Total number of limit candidates evaluated across all pairs.

n_feasible Number of candidates satisfying global feasibility.

n_failed_pairs Number of (n1, n2) pairs where dsnp_limits() failed.

failures A data.frame of failed pairs with columns n1, n2, and message. Empty if no failures occurred.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Joekes, S., Smrekar, M. and Barbosa, E. P. (2015). Extending a double sampling control chart for non-conforming proportion in high quality processes to the case of small samples. Statistical Methodology, 23, 35-49. doi:10.1016/j.stamet.2014.09.003.

See Also

[dsnp_limits](#), [dsnp_prob_accept](#), [dsnp_arl](#), [dsnp_ass](#)

Examples

```
# Small example for fast execution
res <- dsnp_design(
  p0 = 0.05, p1 = 0.10,
  n1_range = 5:6, n2_range = 8:10,
  arl0_min = 50, objective = "arl1"
)
res$best[, c("n1", "n2", "wl", "ucl1", "ucl2", "arl0", "arl1")]
```

dsnp_limits

Double-Sampling np Chart: Limit Search

Description

Search for feasible fractional control limits for the double-sampling np chart by enumerating integer threshold combinations and evaluating each candidate with the shared numerical core used by dsnp_prob_accept(), dsnp_arl(), and dsnp_ass(). This function searches limits for fixed sample sizes; it does not optimize the complete sampling design.

Usage

```
dsnp_limits(
  p0,
  n1,
  n2,
  alpha = 0.0027,
  p1 = NULL,
  conservative = TRUE,
  allow_empty_warning = FALSE,
  max_results = 20,
  curtailed = FALSE
)
```

Arguments

p0	In-control nonconforming proportion, a finite scalar in $[0, 1]$.
n1	First-stage sample size, a positive integer.
n2	Second-stage sample size, a positive integer.
alpha	Maximum desired false-alarm probability, a finite scalar in $(0, 1)$.
p1	Optional out-of-control proportion, a finite scalar in $[0, 1]$. When supplied, ARL1 is the primary performance ranking.
conservative	Logical. If TRUE, candidates satisfying $p_{\text{signal}0} \leq \alpha$ are ranked before infeasible candidates.
allow_empty_warning	Logical. If FALSE, discard candidates with no integer count in the warning zone.
max_results	Positive integer maximum number of ranked candidates to retain.
curtailed	Logical. If TRUE, use curtailed (truncated) inspection ASS instead of complete second-sample ASS. Default FALSE.

Details

The function enumerates valid combinations of `wl_accept`, `ucl1_reject`, and `ucl2_accept`, then converts them to fractional limits. A candidate is feasible when its in-control signal probability is no greater than `alpha`.

With `p1`, candidates are ranked by feasibility when `conservative = TRUE`, then by increasing ARL1, ASS0, and distance from `alpha`. Without `p1`, they are ranked by feasibility, distance from `alpha`, ASS0, and ARL0. Setting `conservative = FALSE` removes feasibility from the front of the corresponding ordering.

When `ucl1_reject = wl_accept + 1`, the warning zone is empty and the chart degenerates to a single-sample scheme. Such candidates are excluded by default; when allowed, their probabilities are computed directly with `stats::pbinom()`.

Value

A list with the following elements:

best The first row of candidates, retained as a data frame.

candidates A data frame containing up to `max_results` ranked candidates. Columns are `wl`, `ucl1`, `ucl2`, `wl_accept`, `ucl1_reject`, `ucl2_accept`, `pt0`, `p_signal0`, `arl0`, `ass0`, and `feasible`; when `p1` is supplied, `pt1`, `p_signal1`, `arl1`, and `ass1` are also included.

p0, p1, n1, n2, alpha The validated input parameters.

conservative, allow_empty_warning, curtailed The validated input flags.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Joeke, S., Smrekar, M. and Barbosa, E. P. (2015). Extending a double sampling control chart for non-conforming proportion in high quality processes to the case of small samples. *Statistical Methodology*, 23, 35–49.

See Also

[dsnp_prob_accept](#), [dsnp_arl](#), [dsnp_ass](#)

Examples

```
limits0 <- dsnp_limits(0.05, 5, 10, alpha = 0.05)
limits0$best
nrow(limits0$candidates)

limits1 <- dsnp_limits(0.05, 5, 10, alpha = 0.05, p1 = 0.10)
limits1$best[, c("wl", "ucl1", "ucl2", "arl0", "arl1", "ass0")]
```

dsnp_prob_accept

Double-Sampling np Chart: Acceptance Probability

Description

Compute the total probability that the double-sampling np chart accepts (does not signal) at a given nonconforming proportion.

Usage

```
dsnp_prob_accept(p, n1, n2, wl, ucl1, ucl2)
```

Arguments

<code>p</code>	Nonconforming proportion to evaluate, a finite numeric scalar or vector in $[0, 1]$.
<code>n1</code>	First-stage sample size, a positive integer.
<code>n2</code>	Second-stage sample size, a positive integer.
<code>wl</code>	Finite fractional warning limit.
<code>uc11</code>	Finite fractional first-stage upper control limit greater than <code>wl</code> .
<code>uc12</code>	Finite fractional combined upper control limit greater than <code>wl</code> .

Details

A subgroup is accepted immediately when the first-stage count D_1 is at or below `floor(wl)` and signals immediately when D_1 is at or above `floor(uc11) + 1`. Counts between those thresholds continue to the second stage. A continued subgroup is accepted when $D_1 + D_2$ is at or below `floor(uc12)` and signals otherwise.

Value

A list with the following elements:

- pa1** First-stage acceptance probability.
- pa2** Second-stage acceptance probability.
- pt** Total acceptance probability, `pa1 + pa2`.
- p_signal** Total signal probability, `1 - pt`.
- p_decision_first** Probability of either accepting or signaling at the first stage.
- p_second** Probability that the second sample is required.
- n1, n2, wl, uc11, uc12** The validated chart parameters.
- wl_accept** Integer threshold `floor(wl)`; accept at stage one when D_1 does not exceed this value.
- uc11_reject** Integer threshold `floor(uc11) + 1`; signal at stage one when D_1 is at least this value.
- uc12_accept** Integer threshold `floor(uc12)`; accept at stage two when $D_1 + D_2$ does not exceed this value.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Joekes, S., Smrekar, M. and Barbosa, E. P. (2015). Extending a double sampling control chart for non-conforming proportion in high quality processes to the case of small samples. *Statistical Methodology*, 23, 35-49.

See Also

[dsnp_ar1](#), [dsnp_ass](#), [dsnp_limits](#)

Examples

```
result <- dsnp_prob_accept(
  0.005, n1 = 34, n2 = 162, w1 = 1.5, ucl1 = 2.5, ucl2 = 4.5
)
result$pt
result$p_signal
```

gv_alpha_risk

*False-Alarm Risk for Generalized Variance Charts***Description**

Evaluate the actual in-control probability that the generalized variance statistic crosses limits produced by `gv_limits()`, and report the corresponding zero-state in-control average run length.

Usage

```
gv_alpha_risk(
  n,
  p,
  det_sigma = 1,
  alpha = 0.0027,
  type = c("normal", "cf", "exact", "simulation"),
  side = c("upper", "two.sided"),
  cf_order = 1,
  nsim = 2e+05,
  seed = NULL
)
```

Arguments

<code>n</code>	Integer subgroup sample size. It must satisfy $n > p$.
<code>p</code>	Integer process dimension, at least 2.
<code>det_sigma</code>	Positive finite scalar equal to $ \Sigma $, the determinant of the in-control covariance matrix. Control limits scale linearly with this value.
<code>alpha</code>	Nominal probability of a false alarm per subgroup, strictly between 0 and 1. For <code>side = "upper"</code> , all probability is placed in the upper tail. For <code>side = "two.sided"</code> , $\alpha / 2$ is placed in each tail.
<code>type</code>	Character string selecting the limit calculation: "normal" Moment-matched Gaussian quantiles using the exact mean and variance of $ S $. "cf" Cornish-Fisher corrected Gaussian quantiles using skewness, and optionally excess kurtosis. "exact" Closed-form chi-square quantiles for $p = 2$, or the published upper-limit table for $p = 3$, $n = 4, \dots, 15$, and α equal to 0.0020 or 0.0027.

	"simulation" Monte Carlo quantiles from the Bartlett product-of-chi-squares representation.
	Partial matching is not used.
side	Either "upper" or "two.sided". Since generalized variance is nonnegative, the lower limit for an upper chart is zero. Any negative approximated lower limit for a two-sided chart is also truncated to zero.
cf_order	Integer 1 or 2. Order 1 uses the skewness correction. Order 2 additionally uses excess kurtosis and the squared-skewness term. It is used only when type = "cf".
nsim	Integer number of Monte Carlo draws, at least 1000. It is used by type = "simulation" and stored in the returned object for all methods.
seed	NULL or a finite numeric scalar used as the Monte Carlo seed. A supplied seed makes simulated limits reproducible and the caller's existing .Random.seed is restored on exit.

Details

The diagnostic is useful because Gaussian and Cornish-Fisher limits are approximations and therefore need not attain the requested nominal risk exactly. For dimension $p = 2$, the crossing probability is evaluated from the exact chi-square representation of $|S|$. For $p > 2$, it is estimated independently by Monte Carlo simulation from Bartlett's product-of-chi-squares representation.

The function first calls `gv_limits()` using the requested type. It then evaluates

$$P(|S| > UCL)$$

for an upper chart, or

$$P(|S| < LCL) + P(|S| > UCL)$$

for a two-sided chart. The inequalities are strict, matching the signaling rule used by `cchart.GV()`.

For $p > 2$, `nsim` controls both any simulation used to construct the limits and the independent simulation used to evaluate their risk. If `seed` is supplied, the limits use that seed and the risk evaluation uses `seed + 1`; each seeded simulation restores the caller's previous random-number state on exit. With `seed = NULL`, the global RNG state advances normally.

Value

A list with components:

`alpha` Actual or Monte Carlo estimated false-alarm probability.

`nominal_alpha` The requested nominal value.

`ar10` Zero-state in-control average run length, $1 / \alpha$; Inf if no simulated crossing is observed.

`limits` The complete object returned by `gv_limits()`.

`method` Either "exact evaluation" for $p = 2$ or "simulation" for higher dimensions.

Monte Carlo uncertainty

For $p > 2$, the reported risk is a binomial proportion based on `nsim` draws. Its approximate Monte Carlo standard error is $\sqrt{\hat{\alpha}(1 - \hat{\alpha})/nsim}$. Very small false-alarm risks therefore require substantially more simulations than routine examples.

Errors

Input and exact-method restrictions are inherited from `gv_limits()`. In addition, simulation requires an integer `nsim` of at least 1000 and a valid optional seed.

References

Barbosa, E. P., Gneri, M. A., and Meneguetti, A. *Improving Shewhart-type Generalized Variance Control Charts for Multivariate Process Variability Monitoring using Cornish-Fisher Quantile Correction, Meijer-G Function and Other Tools*. Research report.

See Also

[gv_limits](#), [gv_stat](#), [cchart.GV](#)

Examples

```
# Exact evaluation shows that exact p = 2 limits attain the nominal risk.
exact_risk <- gv_alpha_risk(10, 2, type = "exact")
exact_risk$alpha
exact_risk$ar10

# Diagnose the false-alarm inflation of moment-matched normal limits.
gv_alpha_risk(10, 2, type = "normal")

# Reproducible risk evaluation for a higher-dimensional chart.
gv_alpha_risk(8, 3, type = "cf", nsim = 10000, seed = 2026)
```

gv_limits

Generalized Variance Control Limits

Description

Compute one-sided upper or equal-tail two-sided control limits for the generalized variance statistic $|S|$ under independent sampling from a p -variate normal process with in-control covariance matrix Σ .

Usage

```
gv_limits(
  n,
  p,
  det_sigma = 1,
  alpha = 0.0027,
  type = c("normal", "cf", "exact", "simulation"),
  side = c("upper", "two.sided"),
  cf_order = 1,
```

```

    nsim = 1e+05,
    seed = NULL
)
```

Arguments

n	Integer subgroup sample size. It must satisfy $n > p$.
p	Integer process dimension, at least 2.
det_sigma	Positive finite scalar equal to $ \Sigma $, the determinant of the in-control covariance matrix. Control limits scale linearly with this value.
alpha	Nominal probability of a false alarm per subgroup, strictly between 0 and 1. For side = "upper", all probability is placed in the upper tail. For side = "two.sided", alpha / 2 is placed in each tail.
type	Character string selecting the limit calculation: "normal" Moment-matched Gaussian quantiles using the exact mean and variance of $ S $. "cf" Cornish-Fisher corrected Gaussian quantiles using skewness, and optionally excess kurtosis. "exact" Closed-form chi-square quantiles for $p = 2$, or the published upper-limit table for $p = 3, n = 4, \dots, 15$, and alpha equal to 0.0020 or 0.0027. "simulation" Monte Carlo quantiles from the Bartlett product-of-chi-squares representation. Partial matching is not used.
side	Either "upper" or "two.sided". Since generalized variance is nonnegative, the lower limit for an upper chart is zero. Any negative approximated lower limit for a two-sided chart is also truncated to zero.
cf_order	Integer 1 or 2. Order 1 uses the skewness correction. Order 2 additionally uses excess kurtosis and the squared-skewness term. It is used only when type = "cf".
nsim	Integer number of Monte Carlo draws, at least 1000. It is used by type = "simulation" and stored in the returned object for all methods.
seed	NULL or a finite numeric scalar used as the Monte Carlo seed. A supplied seed makes simulated limits reproducible and the caller's existing .Random.seed is restored on exit.

Details

If S is the usual covariance matrix from a subgroup of size n , then $(n - 1)S$ has a Wishart distribution and $|S|$ can be represented as a scaled product of independent chi-square variables. The function uses that representation to obtain moments, approximations, exact cases, or simulated quantiles.

Value

A list with components:

lcl, ucl Lower and upper control limits.

center The exact in-control mean of $|S|$; this is the center line used by `cchart.GV()`.

type, side, alpha The selected method, chart sidedness, and nominal false-alarm probability.

n, p, det_sigma The validated design parameters.

moments A list containing the first four ordinary moments, mean, variance, standard deviation, skewness, excess kurtosis, the constants b1 and b2, and the design parameters.

cf_order, nsim, seed The requested numerical settings.

Exact-method scope

The $p = 2$ exact distribution is available for both upper and two-sided charts. The $p = 3$ implementation is deliberately limited to the published one-sided upper quantiles described above. No generic Meijer-G evaluator is claimed; use `type = "simulation"` outside those cases.

Decision convention

A plotted subgroup signals when $|S| < LCL$ or $|S| > UCL$. Equality to either limit is treated as in control.

Errors

Errors are raised for invalid dimensions, $n \leq p$, a nonpositive `det_sigma`, invalid `alpha`, unsupported exact cases, invalid Cornish-Fisher order, fewer than 1000 simulations, or an invalid seed.

References

- Barbosa, E. P., Gneri, M. A., and Meneguetti, A. *Improving Shewhart-type Generalized Variance Control Charts for Multivariate Process Variability Monitoring using Cornish-Fisher Quantile Correction, Meijer-G Function and Other Tools*. Research report.
- Anderson, T. W. (1984). *An Introduction to Multivariate Statistical Analysis*, 2nd ed. Wiley.
- Cornish, E. A. and Fisher, R. A. (1960). The percentage points of distributions having known cumulants. *Technometrics*, 2, 209–225.

See Also

[gv_stat](#), [gv_alpha_risk](#), [cchart.GV](#)

Examples

```
# Published dimension-two illustration: |Sigma| = 0.5320, n = 10.
gv_limits(10, 2, det_sigma = 0.5320, type = "normal")$ucl
gv_limits(10, 2, det_sigma = 0.5320, type = "cf")$ucl
gv_limits(10, 2, det_sigma = 0.5320, type = "exact")$ucl

# Equal-tail exact limits are also available in dimension two.
gv_limits(10, 2, side = "two.sided", type = "exact")
```

```
# A reproducible simulation-based limit for higher dimension.
gv_limits(8, 3, type = "simulation", nsim = 5000, seed = 2026)
```

gv_stat

Generalized Variance by Subgroup

Description

Compute the generalized variance, defined as the determinant of the unbiased sample covariance matrix, for each multivariate subgroup.

Usage

```
gv_stat(x, size = NULL)
```

Arguments

- | | |
|------|--|
| x | <p>Multivariate subgroup data in one of three forms:</p> <ul style="list-style-type: none"> • a non-empty list whose elements are $n \times p$ numeric matrices; • a numeric three-dimensional array with dimensions subgroup by observation by variable; or • a numeric matrix formed by stacking equal-sized subgroups in consecutive blocks of rows. <p>Every observation must be finite.</p> |
| size | <p>Positive integer subgroup size. It is required when x is a stacked matrix and ignored for list or array input. The number of matrix rows must be an exact multiple of size.</p> |

Details

For subgroup i with n observations on p variables, `gv_stat()` calculates

$$|S_i|, \quad S_i = \frac{1}{n-1} \sum_{j=1}^n (x_{ij} - \bar{x}_i)(x_{ij} - \bar{x}_i)^T.$$

All subgroups must have the same n and p , with $p \geq 2$ and $n > p$. The latter condition permits a nonsingular sample covariance matrix under nondegenerate data, although an observed subgroup may still have determinant zero because of exact linear dependence.

Value

A numeric vector of length equal to the number of subgroups. Element i is $|S_i|$. A determinant that is slightly negative only because of floating-point roundoff is returned as zero.

Errors

An error is raised for unsupported input types, empty or unequal-sized subgroups, non-finite observations, fewer than two variables, $n \leq p$, an invalid matrix size, or a covariance determinant that is genuinely negative.

References

Barbosa, E. P., Gneri, M. A., and Meneguetti, A. *Improving Shewhart-type Generalized Variance Control Charts for Multivariate Process Variability Monitoring using Cornish-Fisher Quantile Correction, Meijer-G Function and Other Tools*. Research report.

Anderson, T. W. (1984). *An Introduction to Multivariate Statistical Analysis*, 2nd ed. Wiley.

See Also

[gv_limits](#), [gv_alpha_risk](#), [cchart.GV](#)

Examples

```
g1 <- cbind(c(0, 1, 2, 3), c(0, 2, 1, 3))
g2 <- cbind(c(1, 2, 4, 7), c(3, 1, 5, 2))

gv_stat(list(g1, g2))
gv_stat(rbind(g1, g2), size = 4)

x <- array(c(g1, g2), dim = c(2, 4, 2))
gv_stat(x)
```

moonroof

Moonroof Installation Defect Data

Description

34 subgroups with defect counts (y_i) and sample sizes (n_i) for moonroof installation. This data set is used for u -chart examples.

Format

A data frame with 34 observations on the following 4 variables.

i Index.

yi The number of defects.

ni The sample size.

ui The proportion of defects.

Source

DeVor, R. E., Chang, T., and Sutherland, J. W. (2007). *Statistical Quality Design and Control*, 2nd ed. Pearson.

Examples

```
data(moonroof)
```

pchart_alpha_risk	<i>Exact False-Alarm Risk for p-Chart Limits</i>
-------------------	--

Description

Compute the actual false-alarm probability of p-chart limits under the in-control binomial model. The calculation respects the discreteness of $X \sim \text{Binomial}(n, p)$ and treats a signal as $X/n < LCL$ or $X/n > UCL$.

Usage

```
pchart_alpha_risk(p, n, lcl, ucl)
```

Arguments

p	In-control nonconforming proportion.
n	Positive integer sample size or vector of sample sizes.
lcl	Lower control limit or vector of limits.
ucl	Upper control limit or vector of limits.

Value

A numeric vector with the exact false-alarm probability for each sample size and pair of limits.

Examples

```
limits <- pchart_limits(0.015, 20, type = "cf2")
pchart_alpha_risk(0.015, 20, limits$lcl, limits$ucl)
```

Description

Compute normal or Cornish-Fisher probability limits for a p chart without constructing a plot. The Cornish-Fisher formulas retain the original proportion scale and correct the normal approximation using binomial cumulants.

Usage

```
pchart_limits(
  p,
  n,
  alpha = ALPHA,
  type = c("normal", "cf1", "cf2"),
  sides = c("two.sided", "upper"),
  truncate = TRUE
)
```

Arguments

p	In-control nonconforming proportion. A scalar strictly between zero and one.
n	Positive integer sample size or vector of sample sizes.
alpha	Nominal false alarm probability. Defaults to 0.0027.
type	Limit method: "normal", "cf1", or "cf2".
sides	Either "two.sided" or "upper".
truncate	Logical. If TRUE, limits are restricted to [0, 1].

Details

type = "cf1" uses the first skewness correction proposed by Winterbottom (1993). type = "cf2" uses the operational two-adjustment limits in equation (8) of Joeke and Barbosa (2013). In particular, the second adjustment is evaluated with the positive normal quantile and is applied with the same sign to both limits. This convention reproduces the published limits and false-alarm risks in Tables 2 and 3.

The article recommends the normal approximation for approximately $np(1 - p) \geq 5$, CF1 for approximately $np(1 - p) \geq 0.25$, and CF2 for approximately $np(1 - p) \geq 0.08$. The returned applicable field reports whether each sample size meets the corresponding recommendation.

Value

A list containing center, lcl, ucl, n, alpha, type, sides, npq, and applicable.

References

Winterbottom, A. (1993). Simple adjustment to improve control limits on attribute charts. *Quality and Reliability Engineering International*.

Joeke, S. and Barbosa, E. P. (2013). An improved attribute control chart for monitoring non-conforming proportion in high quality processes. *Control Engineering Practice*, 21, 407–412. [doi:10.1016/j.conengprac.2012.12.005](https://doi.org/10.1016/j.conengprac.2012.12.005).

Examples

```
pchart_limits(0.015, n = 20, type = "normal")
pchart_limits(0.015, n = 20, type = "cf1")
pchart_limits(0.015, n = 20, type = "cf2")
```

pistonrings

Piston Rings Diameter Data

Description

40 subgroups of size 5 containing diameter measurements of piston rings. This data set is used extensively in statistical process control textbooks and throughout the IQCC package to illustrate $\bar{X}$, R , and S charts.

Format

A data frame with 40 observations on the following 5 variables.

V1 The first measure.

V2 The second measure.

V3 The third measure.

V4 The fourth measure.

V5 The fifth measure.

Source

Montgomery, D. C. (2009). *Introduction to Statistical Quality Control*, 6th ed. Wiley. ISBN 978-0-470-16992-6.

Examples

```
data(pistonrings)
```

plot.cchart.DSnp	<i>Plot a DS-np Control Chart</i>
------------------	-----------------------------------

Description

Plot first-stage and second-stage DS-np decisions on separate panels so that the first-stage count and combined count are shown on their proper scales.

Usage

```
## S3 method for class 'cchart.DSnp'
plot(x, ...)
```

Arguments

x	An object of class "cchart.DSnp".
...	Additional graphical parameters passed to the first panel.

Details

The upper panel displays x_1 , the warning limit, and the first-stage upper control limit. The lower panel displays $x_1 + x_2$ only for observations that reached the second stage, together with the second-stage upper control limit.

Value

Invisible x.

print.cchart.DSnp	<i>Print and Summarize a Double-Sampling np Control Chart</i>
-------------------	---

Description

Display or construct a compact summary of a double-sampling np chart, including sample sizes, decision thresholds, operating characteristics, stage outcomes, and signaled observations.

Usage

```
## S3 method for class 'cchart.DSnp'
print(x, digits = max(3L, getOption("digits") - 3L), ...)

## S3 method for class 'cchart.DSnp'
summary(object, ...)

## S3 method for class 'summary.cchart.DSnp'
print(x, digits = max(3L, getOption("digits") - 3L), ...)
```

Arguments

x	An object of class "cchart.DSnp" or "summary.cchart.DSnp".
digits	Number of significant digits used for numerical output.
...	Additional arguments passed to the summary print method; otherwise currently unused.
object	An object of class "cchart.DSnp".

Value

summary.cchart.DSnp() returns an object of class "summary.cchart.DSnp". The returned list contains call, observations, parameters, limits, performance, stage_counts, and signals. The print methods return their input invisibly.

See Also

[cchart.DSnp](#), [plot.cchart.DSnp](#)

print.cchart.GV	<i>Print and Summarize a Generalized Variance Control Chart</i>
-----------------	---

Description

Display or construct a compact statistical summary of a generalized variance chart, including design dimensions, Phase I and Phase II subgroup counts, control-limit method, covariance source, descriptive statistics, and signaled subgroups.

Usage

```
## S3 method for class 'cchart.GV'
print(x, digits = max(3L, getOption("digits") - 3L), ...)

## S3 method for class 'cchart.GV'
summary(object, ...)

## S3 method for class 'summary.cchart.GV'
print(x, digits = max(3L, getOption("digits") - 3L), ...)
```

Arguments

x	An object of class "cchart.GV" or "summary.cchart.GV".
digits	Number of significant digits used for numerical output.
...	Additional arguments passed to the summary print method; otherwise currently unused.
object	An object of class "cchart.GV".

Value

`summary.cchart.GV()` returns an object of class `"summary.cchart.GV"`. The returned list contains `call`, `dimensions`, `subgroups`, `method`, `limits`, `covariance`, `statistics`, and `signals`. The print methods return their input invisibly.

See Also

[cchart.GV](#), [plot.cchart.GV](#)

r_exact_limits	<i>Exact Probability Limits for the R Chart</i>
----------------	---

Description

Compute exact equal-tail probability control limits for the R chart without constructing a plot. The limits are based on the distribution of the relative range $W = R/\sigma$ under normality, using the Studentized range distribution implemented by `stats::qtukey()`.

Usage

```
r_exact_limits(sigma, n, alpha = ALPHA)
```

Arguments

sigma	Positive finite scalar. The in-control process standard deviation. When sigma is estimated from Phase I data, the returned limits are plug-in estimates and do not incorporate the additional uncertainty of Phase I estimation.
n	Integer subgroup size, at least 2.
alpha	Nominal false-alarm probability per subgroup. Defaults to 0.0027. One half is placed in each tail.

Details

The exact limits are

$$LCL = \sigma F_W^{-1}(\alpha/2; n)$$

and

$$UCL = \sigma F_W^{-1}(1 - \alpha/2; n),$$

where F_W^{-1} is the quantile function of $W = R/\sigma$.

Value

A named list with components:

lcl Lower control limit.

ucl Upper control limit.

center $d_2(n)\sigma$, the expected value of the range under normality.

sigma The supplied sigma.

n The supplied subgroup size.

alpha The supplied nominal false-alarm probability.

Phase convention

These limits are computed for a known or separately estimated sigma. When sigma is estimated from a Phase I reference sample, the limits are plug-in limits and do not account for Phase I sampling variability.

Decision rule

A subgroup range R signals out of control when $R < LCL$ or $R > UCL$. Equality to a limit is treated as in control.

Errors

An error is raised when sigma is NA, NaN, or non-positive; when n is smaller than 2, non-integer, or non-finite; or when alpha is not between 0 and 1.

References

Barbosa, E. P., Gneri, M. A., and Meneguetti, A. (2013). Range control charts revisited: Simpler Tippet-like formulae, its practical implementation, and the study of false alarm. *Communications in Statistics - Simulation and Computation*, 42(2), 247–262. doi:10.1080/03610918.2011.639967.

See Also

[r_shewhart_limits](#), [cchart.R](#), [alpha.risk](#), [d2](#)

Examples

```
# Known-sigma limits for subgroup size n = 5, sigma = 2, alpha = 0.0027
r_exact_limits(sigma = 2, n = 5)

# Custom alpha
r_exact_limits(sigma = 1, n = 10, alpha = 0.01)
```

r_shewhart_limits	<i>Conventional Three-Sigma Limits for the R Chart</i>
-------------------	--

Description

Compute the conventional three-sigma (Shewhart) control limits for the R chart without constructing a plot. The limits are based on the constants $d_2(n)$ and $d_3(n)$, the mean and standard deviation of the relative range $W = R/\sigma$ under normality.

Usage

```
r_shewhart_limits(sigma, n, nsigmas = SIGMA_MULT)
```

Arguments

sigma	Positive finite scalar. The in-control process standard deviation. When estimated from Phase I data, the returned limits are plug-in estimates.
n	Integer subgroup size, at least 2.
nsigmas	Positive numeric scalar. The number of sigma units for the control limits. Defaults to 3.

Details

The conventional limits are

$$LCL = \max\{0, d_2(n) - k d_3(n)\} \sigma$$

and

$$UCL = \{d_2(n) + k d_3(n)\} \sigma,$$

where k is the number of sigma units (default 3). Because the distribution of the relative range is not normal, the actual false-alarm probability of these limits can be substantially larger than the nominal 0.0027 associated with three-sigma limits for a normal variate. Use [alpha.risk](#) to evaluate the exact false-alarm probability.

Value

A named list with components:

lcl Lower control limit (zero-truncated).

ucl Upper control limit.

center $d_2(n)\sigma$, the expected value of the range under normality.

nsigmas The supplied multiplier.

sigma The supplied sigma.

n The supplied subgroup size.

Phase convention

These limits are computed for a known or separately estimated sigma. When sigma is estimated from a Phase I reference sample, the limits are plug-in limits.

Decision rule

A subgroup range R signals out of control when $R < LCL$ or $R > UCL$. Equality to a limit is treated as in control.

Errors

An error is raised when sigma is NA, NaN, or non-positive; when n is smaller than 2, non-integer, or non-finite; or when nsigmas is non-positive.

References

Montgomery, D. C. (2009). *Introduction to Statistical Quality Control*, 6th ed. Wiley.

Barbosa, E. P., Gneri, M. A., and Meneguetti, A. (2013). Range control charts revisited: Simpler Tippet-like formulae, its practical implementation, and the study of false alarm. *Communications in Statistics - Simulation and Computation*, 42(2), 247–262. doi:10.1080/03610918.2011.639967.

See Also

`r_exact_limits`, `cchart.R`, `alpha.risk`, `d2`, `d3`

Examples

```
# Conventional three-sigma limits for n = 5, sigma = 2
r_shewhart_limits(sigma = 2, n = 5)

# Custom multiplier
r_shewhart_limits(sigma = 1, n = 10, nsigmas = 2)
```

remove.data

Remove an Observation from Phase I Data

Description

Remove the i -th subgroup from a Phase I data set (array or matrix). Used during Phase I retrospective analysis to eliminate out-of-control signals before recomputing reference limits.

Usage

```
remove.data(datum, i)
```

Arguments

`datum` Phase I data (matrix for $n = 1$, 3D array for $n > 1$).
`i` Index of the subgroup to remove (1-based).

Value

The data set without the i -th subgroup.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

See Also

[data.1](#), [T2.1](#)

Examples

```
mu <- c(5.682, 88.22)
Sigma <- miscTools::symMatrix(c(3.770, -5.495, 13.53), 2)
datum <- data.1(20, 10, mu, Sigma)
# Removing the observatiob 13 from the data set "datum" and updating it:
datum <- remove.data(datum, 13)
```

s_exact_limits	<i>Exact Probability Limits for the S Chart</i>
----------------	---

Description

Compute exact probability control limits for subgroup standard deviations without constructing a plot. Under normal sampling,

$$(n-1)S^2/\sigma^2 \sim \chi_{n-1}^2.$$

For a two-sided chart, the nominal false-alarm probability is divided equally between the two tails. For an upper chart, all of alpha is assigned to the upper tail and the lower limit is zero.

Usage

```
s_exact_limits(sigma, n, alpha = ALPHA, side = c("two.sided", "upper"))
```

Arguments

`sigma` Positive finite scalar. The in-control process standard deviation. When estimated from Phase I data, the returned limits are plug-in limits and do not incorporate Phase I estimation uncertainty.
`n` Integer subgroup size(s), each at least 2.
`alpha` Nominal false-alarm probability per subgroup, strictly between 0 and 1. Defaults to 0.0027.
`side` Either "two.sided" or "upper".

Value

A named list with components `lcl`, `ucl`, `center`, `sigma`, `n`, `alpha`, `side`, and `method`. The theoretical center is $c_4(n)\sigma$.

Phase convention

These functions take `sigma` as known. If a Phase I estimate is supplied instead, the resulting limits are plug-in limits. The historical `cchart.S()` wrapper estimates `sigma` from the data using `qcc::sd.S()` before calling this function for exact limits.

Decision rule

A subgroup standard deviation signals out of control when it is below `lcl` or above `ucl`. Equality to a limit is treated as in control.

References

Montgomery, D. C. (2009). *Introduction to Statistical Quality Control*, 6th ed. Wiley.

See Also

[s_shewhart_limits](#), [cchart.S](#), [c4](#)

Examples

```
s_exact_limits(sigma = 2, n = 5)
s_exact_limits(sigma = 1, n = 2:6, side = "upper")
```

s_shewhart_limits

Conventional Shewhart Limits for the S Chart

Description

Compute conventional Shewhart limits for subgroup standard deviations without constructing a plot. Under normality, $E(S) = c_4(n)\sigma$ and $SD(S) = \sigma\sqrt{1 - c_4(n)^2}$. With the default theoretical center, the limits are therefore

$$LCL = \max\{0, c_4(n)\sigma - k\sigma\sqrt{1 - c_4(n)^2}\}$$

and

$$UCL = c_4(n)\sigma + k\sigma\sqrt{1 - c_4(n)^2},$$

where k is `nsigmas`.

Usage

```
s_shewhart_limits(
  sigma,
  n,
  nsigmas = SIGMA_MULT,
  side = c("two.sided", "upper"),
  center = NULL
)
```

Arguments

sigma	Positive finite scalar. The in-control process standard deviation, known or estimated separately.
n	Integer subgroup size(s), each at least 2.
nsigmas	Positive finite scalar giving the number of standard-error units used for the limits. Defaults to 3.
side	Either "two.sided" or "upper". For an upper chart the lower limit is zero.
center	Optional finite numeric center. It may be a scalar or have the same length as n. If omitted, $c_4(n)\sigma$ is used.

Details

The optional center argument exists to reproduce the historical qcc S-chart convention, which centers the limits on the observed weighted mean of subgroup standard deviations while estimating sigma separately. When center = NULL, the theoretical center $c_4(n)\sigma$ is used.

Value

A named list with components lcl, ucl, center, sigma, n, nsigmas, side, and method.

Legacy qcc convention

The current `qcc::limits.S()` implementation computes `std.dev * sqrt(1 - c4(sizes)^2)` as the standard error of the plotted S statistic and centers conventional limits on the chart center supplied by qcc. Consequently, `cchart.S(type = "n")` passes its Phase I weighted S center explicitly to this function. This preserves the historical IQCC/qcc numerical behavior rather than silently replacing the sample center by $c_4(n)\hat{\sigma}$.

References

Montgomery, D. C. (2009). *Introduction to Statistical Quality Control*, 6th ed. Wiley.

Scrucca, L. *qcc: Quality Control Charts*. R package.

See Also

[s_exact_limits](#), [cchart.S](#), [c4](#)

Examples

```
s_shewhart_limits(sigma = 2, n = 5)
s_shewhart_limits(sigma = 2, n = 5, center = 1.9)
```

sim_t2_asymptotic	<i>Simulate Asymptotic Distribution of Hotelling T^2</i>
-------------------	---

Description

Monte Carlo simulation to study the convergence of the Hotelling T^2 statistic to its asymptotic χ_p^2 distribution under various multivariate distributions. Used to validate the asymptotic robustness result of Gneri and Barbosa (2006).

Usage

```
sim_t2_asymptotic(
  n = c(10, 30, 100, 500),
  p = c(2, 5),
  distributions = c("normal", "t5", "gamma2"),
  nsim = 10000,
  sig_levels = c(0.9, 0.95, 0.99),
  seed = 42,
  rho = 0.3
)
```

Arguments

n	Integer vector of sample sizes to evaluate. Each element must satisfy $n > p$ for the corresponding dimension. Default is <code>c(10, 30, 100, 500)</code> .
p	Integer vector of dimensions to evaluate. Default is <code>c(2, 5)</code> .
distributions	Character vector of distribution names. Available options: "normal" Multivariate normal $N(0, \Sigma)$. All moments finite. Baseline case. "t5" Elliptical multivariate t with 5 degrees of freedom, scaled to covariance Σ . Symmetric, heavy-tailed, finite fourth moment ($4 < 5$). "gamma2" Independent Gamma(2, 1) margins, centered and scaled to covariance Σ . Asymmetric, all moments finite. "t4" Elliptical multivariate t with 4 degrees of freedom (stress test). Has infinite fourth moment, violating the formal condition of Theorem 3.
nsim	Number of Monte Carlo replications. A single positive integer. Default is 10000.
sig_levels	Numeric vector of quantile levels in $(0, 1)$ to compare. Default is <code>c(0.90, 0.95, 0.99)</code> .
seed	Random seed for reproducibility. A single integer. Default is 42.
rho	Correlation parameter for the equicorrelation matrix $\Sigma_{ij} = 1$ if $i = j$, ρ otherwise. Must satisfy $\rho > -1/(p - 1)$ for each p to ensure positive definiteness. Default is 0.3.

Details

For each combination of sample size n , dimension p , and distribution, the function generates `nsim` samples, computes the Hotelling T^2 statistic, and compares empirical quantiles against the limiting χ_p^2 distribution.

The T^2 statistic is computed from a single sample $X_1, \dots, X_n$ as $T^2 = n(\bar{X} - \mu)'S^{-1}(\bar{X} - \mu)$ where $\mu = 0$ is the known null mean and S is the sample covariance. This matches the form in Theorem 3 of Gneri and Barbosa (2006). It does **not** match the Phase I statistic [T2.1](#), which uses estimated grand means and pooled covariance from multiple subgroups.

Value

A data frame with columns:

`n` Sample size.

`p` Dimension.

`distribution` Distribution name.

`level` Quantile level (e.g., 0.95).

`empirical` Empirical quantile of simulated T^2 .

`chisq` Theoretical χ_p^2 quantile.

`mcse` Empirical Monte Carlo standard error estimate for the simulated quantile.

`nsim` Number of valid replications (after discarding singular covariance draws).

RNG preservation

The function saves and restores `.Random.seed` on exit, so it does not alter the global RNG state.

Monte Carlo standard error

The MCSE is estimated from the spacing of simulated order statistics. For probability q , an approximate 95 percent binomial interval for the quantile rank is mapped back to the ordered simulated T^2 values; the interval width divided by $2z_{0.975}$ estimates the standard error. This calculation is distribution-free at the rank level and does not assume that the simulated statistic already follows χ_p^2 .

References

Gneri, M. A. and Barbosa, E. P. (2006). "Robustez Asintótica de la Estadística de Hotelling". Sección 4.2, Teorema 3, pp. 34-36.

Examples

```
# Quick test with few replications
res <- sim_t2_asymptotic(
  n = c(30, 100),
  p = 2,
  distributions = c("normal", "t5"),
  nsim = 500,
```

```
    seed = 42
  )
  res
```

softdrink

Soft Drink Bottling Volume Data

Description

15 subgroups of size 10 containing volume measurements of soft drink bottles. This data set is used throughout the IQCC package for multivariate and univariate control chart examples.

Format

A data frame with 15 lines and 10 columns.

x1 The first measure.

x2 The second measure.

x3 The third measure.

x4 The fourth measure.

x5 The fifth measure.

x6 The sixth measure.

x7 The seventh measure.

x8 The eighth measure.

x9 The ninth measure.

x10 The tenth measure.

Source

Montgomery, D. C. (2009). *Introduction to Statistical Quality Control*, 6th ed. Wiley.

Examples

```
data(softdrink)
```

stats

*Auxiliary Statistics for Hotelling T^2 Charts***Description**

Compute the reference statistics needed by [T2.1](#) and [T2.2](#) from Phase I data. Returns a list with: `[[1]]` grand mean vector ($\bar{\bar{x}}$), `[[2]]` pooled covariance matrix ($\bar{S}$), `[[3]]` matrix of subgroup means.

Usage

```
stats(datum, m, n, p)
```

Arguments

datum	For $n = 1$: $m \times p$ matrix. For $n > 1$: $n \times p \times m$ array.
m	Number of subgroups.
n	Subgroup size.
p	Dimension.

Details

To use this function it is necessary to have the information from [data.1](#).

Value

A list with three components: grand mean (vector), pooled covariance (matrix), subgroup means ($m \times p$ matrix).

Phase convention

Designed for Phase I reference data.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

See Also

[T2.1](#), [data.1](#)

Examples

```
mu <- c(5.682, 88.22)
Sigma <- miscTools::symMatrix(c(3.770, -5.495, 13.53), 2)
#Example with individual observations
datum <- data.1(50, 1, mu, Sigma)
estat <- stats(datum, 50, 1, 2)
#Example with sub-group observations
datum <- data.1(20, 10, mu, Sigma)
estat <- stats(datum, 20, 10, 2)
```

T2.1

Hotelling T² Statistic for Phase I

Description

Calculate the Hotelling T² statistic for Phase I using estimated mean vector and covariance matrix. For individual observations ($n = 1$): $T^2 = x_i' S^{-1} x_i$. For subgroups ($n > 1$): $T_i^2 = n(\bar{x}_i - \bar{\bar{x}})' S^{-1} (\bar{x}_i - \bar{\bar{x}})$.

Usage

```
T2.1(estat, m, n)
```

Arguments

estat	A list returned by <code>stats</code> with three components: <code>[[1]]</code> grand mean vector, <code>[[2]]</code> pooled covariance matrix, <code>[[3]]</code> matrix of subgroup means ($m \times p$).
m	Number of subgroups (Phase I sample size).
n	Subgroup size. Use $n = 1$ for individual observations, $n > 1$ for subgroups.

Details

Before using this function it is necessary to execute `stats` (that calculates the auxiliary statistics involved in the T² formula) and the function `data.1` (or other way to supply the data).

Value

A numeric vector of length m with the T² statistics.

Phase convention

Phase I --- uses parameters estimated from the same data being plotted. The T² statistics are not independent.

Errors

Stop if $n < 1$.

Decision rule

A Phase I point signals when T^2 exceeds the UCL from [cchart.T2.1](#).

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Montgomery, D.C., (2009). "Introduction to Statistical Quality Control". Chapter 11. Wiley.

See Also

[T2.2](#), [cchart.T2.1](#), [stats](#), [data.1](#), [robustez](#)

Examples

```
mu <- c(5.682, 88.22)
Sigma <- miscTools::symMatrix(c(3.770, -5.495, 13.53), 2)
#Example with individual observations
datum <- data.1(50, 1, mu, Sigma)
estat <- stats(datum, 50, 1, 2)
T2.1(estat, 50, 1)
#Example with sub group observations
datum <- data.1(20, 10, mu, Sigma)
estat <- stats(datum, 20, 10, 2)
T2.1(estat, 20, 10)
```

T2.2

Hotelling T^2 Statistic for Phase II

Description

Calculate T^2 for new (Phase II) observations using Phase I reference parameters. For individual observations: $T^2 = (x_{new} - \bar{x})' S^{-1} (x_{new} - \bar{x})$. For subgroups: $T^2 = n(\bar{x}_{new} - \bar{\bar{x}})' S^{-1} (\bar{x}_{new} - \bar{\bar{x}})$.

Usage

```
T2.2(datum2, estat, n)
```

Arguments

datum2	Phase II data. For $n = 1$: a numeric vector of length p . For $n > 1$: an $n \times p$ matrix.
estat	List from stats (Phase I estimates).
n	Subgroup size. Use $n = 1$ for individual observations, $n > 1$ for subgroups.

Details

Before using this function it is necessary to execute [stats](#) (that calculates the auxiliary statistics involved in the T^2 formula) and the function [data.2](#) (or other way to supply the data).

Value

A scalar T^2 statistic.

Phase convention

Phase II --- tests new observations against Phase I reference distribution.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Montgomery, D.C., (2009). "Introduction to Statistical Quality Control". Chapter 11. Wiley.

See Also

[T2.1](#), [cchart.T2.2](#), [robustez](#)

Examples

```
mu <- c(5.682, 88.22)
Sigma <- miscTools::symMatrix(c(3.770, -5.495, 13.53), 2)
#Example with individual observations
datum <- data.1(50, 1, mu, Sigma)
estat <- stats(datum, 50, 1, 2)
datum2 <- data.2(estat, 1, p = 2)
T2II <- T2.2(datum2, estat, 1)
#Example with subgroup observations
datum <- data.1(20, 10, mu, Sigma)
estat <- stats(datum, 20, 10, 2)
datum2 <- data.2(estat, 10, p = 2)
T2II <- T2.2(datum2, estat, 10)
```

table.const	Table of Control Chart Constants d_2 , d_3 , and c_4
-------------	--

Description

Build a matrix with the constants d_2 , d_3 , and c_4 for sample sizes $n = 2, 3, \dots$, the specified maximum.

Usage

```
table.const(n)
```

Arguments

n Maximum sample size (must be ≥ 2). The table includes rows for $n = 2, 3, \dots, n$.

Details

The three constants are those returned by [d2](#), [d3](#), and [c4](#) respectively. They are the standard She-whart chart constants for the relative range and the standard deviation under normality.

Value

A matrix with $n - 1$ rows and 3 columns named "d2", "d3", "c4". Row names are the sample sizes.

Errors

Stops if $n < 2$, raised by the internal [d2](#) call.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Montgomery, D. C. (2009). *Introduction to Statistical Quality Control*, 6th ed. Wiley.

See Also

[d2](#), [d3](#), [c4](#)

Examples

```
table.const(10)
table.const(5)
```

table.qtukey	<i>Tukey Quantile Table for the Relative Range</i>
--------------	--

Description

Print a table of quantiles of the studentized range distribution $W = R/\sigma$ for sample sizes $n = 2$ through the specified maximum, using [qtukey](#) with infinite denominator degrees of freedom.

Usage

```
table.qtukey(alpha, n)
```

Arguments

alpha	Type-I error probability ($0 < \alpha < 1$).
n	Maximum sample size (≥ 2).

Details

Four tail probabilities are tabulated for each n : $\alpha/2$, α , $1 - \alpha$, and $1 - \alpha/2$. The default examples use $\alpha = 0.0027$ (US convention) and $\alpha = 0.0020$ (European convention).

Value

Invisibly, a matrix with $n - 1$ rows and 4 columns. The matrix is also printed to the console.

Errors

Stops if $n < 2$ or if alpha is not strictly between 0 and 1.

Author(s)

Daniela R. Recchia, Emanuel P. Barbosa

References

Barbosa, E. P., Gneri, M. A. and Meneguetti, A. (2013). On the the evaluation of the relative range distribution. *Communications in Statistics - Simulation and Computation*, 42, 1311–1339. [doi:10.1080/03610918.2011.639967](https://doi.org/10.1080/03610918.2011.639967).

See Also

[alpha.risk](#), [cchart.R](#)

Examples

```
table.qtukey(0.0027, 15)
table.qtukey(0.0020, 10)
```

trv_alpha_risk

False-Alarm Risk for Trace-Statistic Charts

Description

Evaluate the actual in-control upper-tail probability of a supplied $\text{tr}(V)$ upper control limit.

Usage

```
trv_alpha_risk(
  n,
  p,
  ucl,
  Sigma0 = NULL,
  type = c("chisq", "simulation"),
  nsim = 1e+05,
  seed = NULL
)
```

Arguments

<code>n</code>	Integer subgroup sample size, at least 2.
<code>p</code>	Integer process dimension, at least 2.
<code>ucl</code>	Positive finite upper control limit on the trace statistic.
<code>Sigma0</code>	Optional in-control covariance matrix. It is accepted for API symmetry with trv_stat and validated when supplied, but the null distribution of the standardized trace statistic depends only on <code>n</code> and <code>p</code> .
<code>type</code>	Character string selecting the calculation: "chisq" Exact chi-square quantile with $p(n - 1)$ degrees of freedom. "simulation" Monte Carlo quantile from the same chi-square distribution. This is mainly a diagnostic and teaching option.
<code>nsim</code>	Integer number of Monte Carlo draws, at least 1000. Used only when <code>type = "simulation"</code> .
<code>seed</code>	NULL or a finite numeric scalar used as the Monte Carlo seed. A supplied seed makes simulated limits reproducible and restores the caller's existing random-number state on exit.

Value

A list with components `alpha`, `ar10`, `ucl`, `n`, `p`, `df`, and `method`.

Examples

```
lim <- trv_limits(8, 2)
trv_alpha_risk(8, 2, lim$ucl)
```

trv_limits

*Trace-Statistic Control Limits***Description**

Compute one-sided upper control limits for the auxiliary trace statistic $T = (n - 1) \text{tr}(\Sigma_0^{-1} S)$.

Usage

```
trv_limits(
  n,
  p,
  alpha = 0.0027,
  type = c("chisq", "simulation"),
  nsim = 1e+05,
  seed = NULL
)
```

Arguments

n	Integer subgroup sample size, at least 2.
p	Integer process dimension, at least 2.
alpha	Nominal upper-tail false-alarm probability, strictly between 0 and 1.
type	Character string selecting the calculation: "chisq" Exact chi-square quantile with $p(n - 1)$ degrees of freedom. "simulation" Monte Carlo quantile from the same chi-square distribution. This is mainly a diagnostic and teaching option.
nsim	Integer number of Monte Carlo draws, at least 1000. Used only when type = "simulation".
seed	NULL or a finite numeric scalar used as the Monte Carlo seed. A supplied seed makes simulated limits reproducible and restores the caller's existing random-number state on exit.

Details

Table 3 of Barbosa, Gneri, and Meneguetti uses N as the row label, although its numerical values correspond to subgroup size n and $p(n - 1)$ degrees of freedom. Its printed probability headings also appear reversed: the column headed 0.9980 matches the 0.9973 quantile and vice versa. This function follows the mathematical definition and evaluates `qchisq(1 - alpha, df = p * (n - 1))`; it does not interchange tail probabilities to reproduce the apparent table-label error.

Value

A list with components `lcl`, `ucl`, `center`, `type`, `alpha`, `n`, `p`, `df`, `nsim`, and `seed`. The lower control limit is zero because the chart is one-sided upper.

Examples

```
trv_limits(n = 8, p = 2)
trv_limits(n = 8, p = 2, type = "simulation", nsim = 5000, seed = 2026)
```

trv_stat

*Trace Statistic for Multivariate Variability***Description**

Compute the auxiliary trace statistic

$$T = (n - 1) \operatorname{tr}(\Sigma_0^{-1} S),$$

where S is the unbiased subgroup covariance matrix and Σ_0 is the in-control covariance matrix.

Usage

```
trv_stat(x, size = NULL, Sigma0)
```

Arguments

<code>x</code>	Multivariate subgroup data in any format accepted by gv_stat : a list of numeric matrices, a subgroup-by-observation-by-variable array, or a stacked numeric matrix.
<code>size</code>	Positive integer subgroup size when <code>x</code> is a stacked matrix. It is ignored for list and array input.
<code>Sigma0</code>	Finite symmetric positive-definite in-control covariance matrix.

Details

The statistic complements the generalized variance $|S|$. Two covariance matrices can have the same determinant but different trace after standardization by Σ_0 ; `trv_stat()` is designed to make that structural change visible.

Under independent sampling from a multivariate normal process with covariance Σ_0 ,

$$(n - 1)S \sim W_p(n - 1, \Sigma_0)$$

and therefore

$$(n - 1) \operatorname{tr}(\Sigma_0^{-1} S) \sim \chi_{p(n-1)}^2.$$

The implementation evaluates the trace as $(n - 1) * \operatorname{sum}(\text{Sigma0_inverse} * S)$, avoiding an explicit matrix square root while preserving the same value.

Value

A numeric vector with one trace statistic per subgroup.

Errors

Errors are raised for invalid subgroup data, non-finite observations, subgroup sizes smaller than two, a non-square or non-symmetric Σ_0 , a dimension mismatch, or a covariance matrix that is not positive definite.

References

Barbosa, E. P., Gneri, M. A., and Meneguetti, A. *Improving Shewhart-type Generalized Variance Control Charts for Multivariate Process Variability Monitoring using Cornish-Fisher Quantile Correction, Meijer-G Function and Other Tools*. Research report.

Anderson, T. W. (1984). *An Introduction to Multivariate Statistical Analysis*, 2nd ed. Wiley.

See Also

[trv_limits](#), [trv_alpha_risk](#), [cchart.trv](#), [gv_stat](#)

Examples

```
g1 <- cbind(c(0, 1, 2, 3), c(0, 2, 1, 3))
g2 <- cbind(c(1, 2, 4, 7), c(3, 1, 5, 2))
trv_stat(list(g1, g2), Sigma0 = diag(2))
```

uchart_alpha_risk	<i>Exact False-Alarm Risk for u-Chart Limits</i>
-------------------	--

Description

Compute the actual false-alarm probability under $X \sim \text{Poisson}(\text{lambda})$, with a signal when $X/n < LCL$ or $X/n > UCL$.

Usage

```
uchart_alpha_risk(lambda, n, lcl, ucl)
```

Arguments

lambda	In-control defect rate per inspection unit.
n	Positive inspection size or vector of inspection sizes.
lcl	Lower control limit or vector of limits.
ucl	Upper control limit or vector of limits.

Value

A numeric vector of exact false-alarm probabilities.

Examples

```
limits <- uchart_limits(1.4, 10, type = "cf2")
uchart_alpha_risk(1.4, 10, limits$lcl, limits$ucl)
```

uchart_limits	<i>Probability Limits for u Charts</i>
---------------	--

Description

Compute normal or Cornish-Fisher probability limits for a u chart without constructing a plot. The model is $X \sim \text{Poisson}(\text{lambda})$ and the monitored rate is $U = X/n$.

Usage

```
uchart_limits(
  lambda,
  n,
  alpha = ALPHA,
  type = c("normal", "cf1", "cf2"),
  sides = c("two.sided", "upper"),
  truncate = TRUE
)
```

Arguments

lambda	In-control defect rate per inspection unit. A positive scalar.
n	Positive inspection size or vector of inspection sizes.
alpha	Nominal false-alarm probability. Defaults to 0.0027.
type	Limit method: "normal", "cf1", or "cf2".
sides	Either "two.sided" or "upper".
truncate	Logical. If TRUE, negative lower limits are set to 0.

Details

For U , the standardized third and fourth cumulants are $\gamma_1 = 1/\sqrt{\text{lambda}}$ and $\gamma_2 = 1/(\text{lambda})$. Consequently, the first Cornish-Fisher correction is $(z^2 - 1)/(6n)$ and the second-order correction reduces to $z(1 - z^2)/(72n\sqrt{\text{lambda}})$. Each tail is evaluated at its signed normal quantile: $z_{\text{upper}} > 0$ for the UCL and $z_{\text{lower}} < 0$ for the LCL.

With $z = 3$, the CF2 limits recover

$$UCL = \text{lambda} + 3\sqrt{\text{lambda}/n} + 4/(3n) - 1/(3n\sqrt{\text{lambda}})$$

and

$$LCL = \text{lambda} - 3\sqrt{\text{lambda}/n} + 4/(3n) + 1/(3n\sqrt{\text{lambda}}),$$

before optional truncation at zero.

Value

A list containing center, lcl, ucl, n, alpha, type, sides, and mean_count.

Examples

```
uchart_limits(1.4, 10, type = "normal")
uchart_limits(1.4, 10, type = "cf1")
uchart_limits(1.4, 10, type = "cf2")
```

Index

add.data, 4
alpha.risk, 5, 16, 54–56, 68

binomdata, 6

c4, 7, 18, 29, 30, 58, 59, 67
cchart.DSnp, 3, 8, 52
cchart.GV, 3, 10, 22, 43, 45, 47, 53
cchart.p, 14
cchart.R, 3, 6, 15, 18, 54, 56, 68
cchart.S, 16, 17, 58, 59
cchart.T2.1, 18, 20, 65
cchart.T2.2, 4, 19, 19, 66
cchart.trV, 21, 72
cchart.u, 23
cchart.Xbar, 24, 26–28
cchart.Xbar1, 25, 26, 26–28
cchart.Xbar2, 25, 26, 27, 27
cchart.Xbar_R, 25, 25, 27

d2, 6, 8, 28, 29, 30, 54, 56, 67
d3, 6, 8, 29, 29, 56, 67
data.1, 30, 31, 57, 63–65
data.2, 31, 31, 66
dsnp_ar1, 10, 32, 34, 37, 39, 40
dsnp_ass, 10, 33, 33, 37, 39, 40
dsnp_design, 35
dsnp_limits, 3, 10, 35, 37, 37, 40
dsnp_prob_accept, 10, 33, 34, 37, 39, 39

gv_alpha_risk, 13, 41, 45, 47
gv_limits, 3, 13, 43, 43, 47
gv_stat, 13, 43, 45, 46, 71, 72

IQQC (IQQC-package), 3
IQQC-package, 3

moonroof, 47

pchart_alpha_risk, 48
pchart_limits, 3, 49

pistonrings, 50
plot.cchart.DSnp, 51, 52
plot.cchart.GV, 53
plot.cchart.GV (cchart.GV), 10
plot.cchart.trV (cchart.trV), 21
print.cchart.DSnp, 51
print.cchart.GV, 13, 52
print.cchart.trV (cchart.trV), 21
print.dsnp_design (dsnp_design), 35
print.summary.cchart.DSnp
 (print.cchart.DSnp), 51
print.summary.cchart.GV
 (print.cchart.GV), 52
print.summary.cchart.trV (cchart.trV),
 21
ptukey, 28, 29

qcc, 25
qtukey, 67

r_exact_limits, 16, 53, 56
r_shewhart_limits, 16, 54, 55
remove.data, 56
robustez, 19, 20, 65, 66

s_exact_limits, 17, 18, 57, 59
s_shewhart_limits, 17, 18, 58, 58
sim_t2_asymptotic, 60
softdrink, 62
stats, 4, 31, 63, 64–66
summary.cchart.DSnp
 (print.cchart.DSnp), 51
summary.cchart.GV (print.cchart.GV), 52
summary.cchart.trV (cchart.trV), 21

T2.1, 18, 19, 31, 57, 61, 63, 64, 66
T2.2, 4, 20, 31, 63, 65, 65
table.const, 8, 29, 66
table.qtukey, 6, 16, 67
trv_alpha_risk, 68, 72

trv_limits, [70](#), [72](#)

trv_stat, [22](#), [69](#), [71](#)

uchart_alpha_risk, [72](#)

uchart_limits, [3](#), [73](#)