Package 'shewhartr'

September 25, 2026

Type Package

Title Statistical Process Control with Tidyverse-Native Workflows

Version 1.4.0

Description A comprehensive toolkit for Statistical Process Control (SPC) that combines the rigor of classical Shewhart methodology with modern tidyverse-native interfaces. Provides classical control charts for variables (I-MR, Xbar-R, Xbar-S) and attributes (p, np, c, u), as well as regression-based control charts for processes with trend. Includes Nelson runs tests, Average Run Length (ARL) simulation, process capability indices with bootstrap confidence intervals, Box-Cox transformation guidance, and a clean Phase I / Phase II workflow. All chart objects integrate with broom via 'tidy', 'glance' and 'augment' methods. References: Shewhart (1931, ISBN:0-87389-076-0); Montgomery (2019, ISBN:978-1-119-39930-8); Nelson (1984) <doi:10.1080/00224065.1984.11978921>; Woodall (2000) <doi:10.1080/00224065.2000.11980013>; Box & Cox (1964) <doi:10.1111/j.2517-6161.1964.tb00553.x>.

License GPL-3

URL https://castlaboratory.github.io/shewhartr/, https://github.com/castlaboratory/shewhartr

BugReports https://github.com/castlaboratory/shewhartr/issues

Encoding UTF-8

LazyData true

Depends R (>= 4.1.0)

Imports rlang (>= 1.1.0), cli (>= 3.6.0), dplyr (>= 1.1.0), purrr (>= 1.0.0), tibble (>= 3.2.0), tidyselect (>= 1.2.0), ggplot2 (>= 3.5.0), broom (>= 1.0.0), slider (>= 0.3.0), grid, grDevices, stats, utils

Suggests patchwork (>= 1.1.0), plotly (>= 4.10.0), qcc (>= 2.7), testthat (>= 3.2.0), knitr, rmarkdown, pkgdown, MASS

Config/testthat/edition 3

VignetteBuilder knitr

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author André Leite [aut, cre],
Hugo Vasconcelos [aut],
Raydonal Ospina [aut],
Cristiano Ferraz [aut],
Castlab [cph, fnd]

Maintainer André Leite <leite@castlab.org>

Repository CRAN

Date/Publication 2026-09-25 15:40:02 UTC

Contents

as_plotly . . . 3
augment.shewhart_chart . . . 4
autoplot.shewhart_chart . . . 5
autoplot.shewhart_regression . . . 6
bacterial_growth . . . 7
bottle_fill . . . 8
box_cox . . . 9
calibrate . . . 10
claims_p . . . 11
color_hue . . . 12
cvd_brazil . . . 12
cvd_recife . . . 14
fit_gompertz_dummy . . . 15
glance.shewhart_chart . . . 15
Gompertz . . . 16
iloglog . . . 17
inv_box_cox . . . 18
is_shewhart_chart . . . 19
loglog . . . 19
monitor . . . 20
pcb_solder . . . 21
print.shewhart_chart . . . 22
rolling_sum . . . 22
shewhart_arl . . . 23
shewhart_box_cox . . . 24
shewhart_c . . . 25
shewhart_capability . . . 26
shewhart_constants . . . 28
shewhart_cusum . . . 28
shewhart_diagnostics . . . 30
shewhart_ewma . . . 31
shewhart_hotelling . . . 33

shewhart_i_mr . . . 34
shewhart_mcusum . . . 36
shewhart_mewma . . . 37
shewhart_np . . . 39
shewhart_p . . . 40
shewhart_palette . . . 42
shewhart_regression . . . 43
shewhart_rules_available . . . 46
shewhart_runs . . . 46
shewhart_theme . . . 47
shewhart_u . . . 48
shewhart_xbar_r . . . 49
shewhart_xbar_s . . . 51
SSgompertzDummy . . . 52
summary.shewhart_chart . . . 53
tablet_weight . . . 54
temperature_drift . . . 54
tidy.shewhart_chart . . . 55

Index **57**

`as_plotly` *Convert a Shewhart chart to an interactive plotly figure*

Description

Produces an interactive HTML plotly version of the chart that `ggplot2::autoplot()` would build for the same object. Useful for dashboards, reports, and any context where hovering, zooming and panning matters.

Usage

```
as_plotly(x, ...)

## Default S3 method:
as_plotly(x, ...)

## S3 method for class 'shewhart_chart'
as_plotly(
  x,
  tooltip = c("x", "y"),
  show_violations = TRUE,
  show_sigma_zones = FALSE,
  locale = NULL,
  ...
)
```

Arguments

x	A shewhart_chart object.
...	Additional arguments forwarded to plotly::ggplotly().
tooltip	Character vector of aesthetics to display in the hover tooltip, as accepted by plotly::ggplotly(). Defaults to c("x", "y").
show_violations	Logical. Highlight violations with red points? Default TRUE.
show_sigma_zones	Logical. Show 1- and 2-sigma zones as shaded bands? Default FALSE.
locale	Optional override for the chart's stored locale.

Details

This is a separate function rather than an autoplot() argument so that loading shewhartr does not pull plotly (and its full transitive dependency tree) into every R session that uses the package. plotly lives in Suggests; install it explicitly if you want to use this function.

Value

A plotly object (S3 class plotly / htmlwidget) ready to print, embed in a Shiny app, or save with htmlwidgets::saveWidget().

Examples

```
if (requireNamespace("plotly", quietly = TRUE)) {
  set.seed(1)
  df  <- data.frame(t = 1:50, y = rnorm(50, mean = 100, sd = 2))
  fit <- shewhart_i_mr(df, value = y, index = t)
  as_plotly(fit)
}
```

---

augment.shewhart_chart

Augment new data with control-chart annotations

Description

Returns the per-observation augmented tibble, optionally re-aligned against fresh data passed via newdata (Phase II monitoring). When newdata is NULL, returns the in-sample augmented tibble.

Usage

```
## S3 method for class 'shewhart_chart'
augment(x, newdata = NULL, ...)
```

Arguments

x	A shewhart_chart object.
newdata	Optional data frame with the same columns as the data used to fit x. If supplied, control limits are propagated to the new rows (Phase II monitoring) and rule violations are re-evaluated.
...	Currently unused.

Value

A tibble. When newdata = NULL, the chart's augmented tibble; otherwise the same shape but for newdata.

Examples

```
set.seed(1)
df <- data.frame(y = rnorm(50))
fit <- shewhart_i_mr(df, value = y)
broom::augment(fit)
```

---

autoplot.shewhart_chart

Plot a Shewhart chart with ggplot2

Description

Generic autoplot method that dispatches on chart subclass. All versions return a ggplot object that the user can further customise with the usual ggplot2 grammar.

Usage

```
autoplot.shewhart_chart(
  object,
  show_violations = TRUE,
  show_sigma_zones = FALSE,
  locale = NULL,
  ...
)
```

Arguments

object	A shewhart_chart object.
show_violations	Logical. Highlight violations with red points? Default TRUE.
show_sigma_zones	Logical. Show 1- and 2-sigma zones as shaded bands? Default FALSE.
locale	Optional override for the chart's stored locale.
...	Passed to subclass-specific methods.

Value

A ggplot object (or, for I-MR / Xbar-R / Xbar-S charts, a list of two ggplot objects with class shewhart_plot_pair that prints them stacked).

Examples

```
fit <- shewhart_i_mr(data.frame(y = rnorm(50)), value = y)
ggplot2::autoplot(fit)
```

---

autoplot.shewhart_regression

Plot a regression control chart

Description

The autoplot() method for shewhart_regression() charts: per phase, a shaded band between dashed limits and a solid centre line; observations coloured by phase; out-of-control points ringed.

Usage

```
autoplot.shewhart_regression(
  object,
  show_violations = TRUE,
  show_sigma_zones = FALSE,
  locale = NULL,
  phase_dates = NULL,
  legend_position = c("top", "inside"),
  ...
)
```

Arguments

object	A shewhart_regression chart.
show_violations	Logical. Ring out-of-control points?
show_sigma_zones	Ignored; kept for a uniform signature.
locale	Optional override for the chart's stored locale.
phase_dates	Show where each phase ends in the legend, e.g. "Phase 1 (until 2020-05-18)"? NULL (default) does so when the index column is a Date or date-time; TRUE forces it for any index (the last index value of each phase); FALSE turns it off. The last phase ("Monitoring") never carries an end.

`legend_position`
: `"top"` (default) puts the legend above the panel, wrapping to several rows when there are many phases; `"inside"` stacks it in a single column in the top-left corner of the panel, on the panel colour, like the figures of Ferraz et al. (2020).

`...`
: Unused.

Value

A `ggplot` object.

Examples

```
pe <- subset(cvd_brazil, region == "PE" & date <= as.Date("2020-06-20"))
fit <- shewhart_regression(pe, value = new_deaths, index = date,
                           model = "log", limits_scale = "model",
                           phase_rule = "we_seven_same")
ggplot2::autoplot(fit, legend_position = "inside")
ggplot2::autoplot(fit, phase_dates = FALSE, locale = "pt")
```

---

bacterial_growth *Bacterial growth curve (optical density)*

Description

A synthetic dataset of optical density (OD) measurements from a bacterial culture, sampled at 80 evenly spaced time points across a 24-hour incubation. The true mean follows a Gompertz growth curve with asymptote 1.2.

Usage

```
bacterial_growth
```

Format

A tibble with 80 rows and 2 columns:

hour Numeric time in hours since inoculation.

od Numeric optical density at 600 nm.

Source

Synthetic. See `data-raw/build_all.R`.

See Also

`shewhart_regression()` with `model = "gompertz"`, `Gompertz()`.

Examples

```
fit <- shewhart_regression(bacterial_growth,
                           value = od, index = hour,
                           model = "gompertz")
ggplot2::autoplot(fit)
```

`bottle_fill` *Bottle filling volumes*

Description

A synthetic dataset of 100 individual fill volumes (in millilitres). Process target is 500 ml with sigma 1.2 ml. A linear drift begins around observation 65, simulating a slowly miscalibrating filler.

Usage

```
bottle_fill
```

Format

A tibble with 100 rows and 2 columns:

observation Integer observation index.

ml Numeric volume in millilitres.

Source

Synthetic. See `data-raw/build_all.R`.

See Also

`shewhart_i_mr()`.

Examples

```
fit <- shewhart_i_mr(bottle_fill, value = ml, index = observation)

ggplot2::autoplot(fit)
```

box_cox *Apply / invert a Box-Cox power transformation*

Description

The Box-Cox transformation (Box & Cox, 1964) is

$$y(\lambda) = \begin{cases} (x^\lambda - 1)/\lambda & \lambda \neq 0 \\ \log(x) & \lambda = 0. \end{cases}$$

Usage

```
box_cox(x, lambda)
```

Arguments

`x`	Numeric vector of strictly positive values.
`lambda`	Numeric scalar. Power parameter.

Details

For `lambda = 0` this returns `log(x)`; for `lambda = 1` it returns `x - 1` (no shape change). Use `shewhart_box_cox()` to estimate `lambda` from the data via profile log-likelihood.

Value

A numeric vector of transformed values.

References

Box, G. E. P., & Cox, D. R. (1964). An Analysis of Transformations. *Journal of the Royal Statistical Society, Series B*, 26(2), 211-252. doi:10.1111/j.25176161.1964.tb00553.x

See Also

`shewhart_box_cox()` to estimate lambda from data.

Examples

```
box_cox(1:10, lambda = 0)      # equivalent to log(1:10)
box_cox(1:10, lambda = 0.5)
```

`calibrate` *Phase I calibration of a control chart*

Description

Convenience wrapper that fits a control chart and tags its phase as `"phase_1"` (the default for any chart constructor). The intent is to make Phase I usage *explicit* in code: the practitioner acknowledges that limits are being estimated.

Usage

```
calibrate(data, ..., chart = "i_mr", trim_outliers = FALSE, max_trim_iter = 5L)
```

Arguments

`data`	A data frame.
`...`	Arguments passed to a chart constructor.
`chart`	A character key naming the chart constructor: `"i_mr"` (default), `"xbar_r"`, `"xbar_s"`, `"p"`, `"np"`, `"c"`, `"u"`, `"regression"`, `"ewma"`, `"cusum"`, `"hotelling"`, `"mewma"`, `"mcusum"`.
`trim_outliers`	Logical. If `TRUE`, iteratively drop observations that violate the rules and re-estimate limits (Montgomery 2019, Section 6.2.3).
`max_trim_iter`	Integer. Maximum trimming iterations.

Details

Optionally drops violations from the in-control estimate ("trimmed" calibration), per the iterative procedure described in Montgomery (2019) Section 6.2.3: if any observation falls outside the limits, it is removed and the limits are recomputed; iterate until either all remaining points are in control or no further trimming is possible.

Value

A shewhart_chart object with `$phase = "phase_1"`.

References

Woodall, W. H. (2000). Controversies and Contradictions in Statistical Process Control. *Journal of Quality Technology*, 32(4), 341-350. doi:10.1080/00224065.2000.11980013

Montgomery, D. C. (2019). *Introduction to Statistical Quality Control* (8th ed.). Wiley. Section 6.2.3.

Examples

```
set.seed(1)
df <- data.frame(y = c(rnorm(40, mean = 100, sd = 2), 110, rnorm(20, 100, 2)))
calib <- calibrate(df, value = y, chart = "i_mr", trim_outliers = TRUE)
calib$n          # 60 if trim was avoided, fewer if outliers were removed
```

---

claims_p *Daily insurance claim error rates*

Description

A synthetic dataset of 30 days. Each day, a variable number of claims is processed (80-150) and the count of claims with errors is recorded. Underlying error rate is 5% for the first 22 days and 9% thereafter.

Usage

```
claims_p
```

Format

A tibble with 30 rows and 3 columns:

day Integer day index (1-30).

n Integer total claims processed that day.

defects Integer claims found with errors.

Source

Synthetic. See `data-raw/build_all.R`.

See Also

`shewhart_p()`.

Examples

```
fit <- shewhart_p(claims_p, defects = defects, n = n, index = day)

ggplot2::autoplot(fit)
```

color_hue *Generate a qualitative HCL palette*

Description

Internal palette used as a default when a number of phases or subgroups need distinct colours. For most plots, ggplot2's default palette is fine; this is provided for backward compatibility and quick prototyping.

Usage

```
color_hue(n)
```

Arguments

n	Integer (>= 1). Number of colours to generate.

Value

A character vector of `n` hex colour strings.

Examples

```
color_hue(5)
```

cvd_brazil *COVID-19 daily mortality, Brazil, Pernambuco and Sao Paulo, 2020*

Description

Daily count of new COVID-19 deaths officially recorded for Brazil as a whole, for the state of Pernambuco and for the state of Sao Paulo, from 25 February 2020 (Pernambuco: 12 March 2020, its first record) to 31 December 2020. These are the series analysed by Ferraz et al. (2020) with the regression chart on the log scale (`shewhart_regression(model = "log", limits_scale = "model")`).

Usage

```
cvd_brazil
```

Format

A tibble with 917 rows and 3 columns, in long format (one row per region and day, sorted by region then date):

date Date of the bulletin.

region Factor with levels `"BR"` (Brazil), `"PE"` (Pernambuco) and `"SP"` (Sao Paulo).

new_deaths Integer count of new deaths reported that day.

Details

Values are kept exactly as published. The Pernambuco series has one negative value, `new_deaths = -37` on 2020-09-03: a bulletin that revised the cumulative total downwards. Log-scale models reject it (`log(1 + y)` is undefined), so filter or reconcile it before fitting a chart that spans that date, e.g. `subset(cvd_brazil, region == "PE" & new_deaths >= 0)`.

The articles used the Ministry of Health bulletins as available on 20 June 2020 (Revista Brasileira de Estatistica) and 24 July 2020 (SBPO). Later bulletins revised some days, so this snapshot can differ slightly from the data behind the published figures, and phase dates may move by a day or two.

Source

Wesley Cota, covid19br: number of confirmed cases and deaths by COVID-19 in Brazil (https://github.com/wcota/covid19br), compiled from the Brazilian Ministry of Health bulletins (https://covid.saude.gov.br). See `data-raw/build_all.R`.

References

Ferraz, C., Petenate, A. J., Leite Wanderley, A., Ospina, R., Torres, J., & Peruzzi Moreira, A. (2020). COVID-19: monitoramento por graficos de Shewhart. *Revista Brasileira de Estatistica*, 78(245), 23-41.

See Also

`shewhart_regression()`, cvd_recife.

Examples

```
table(cvd_brazil$region)

sp <- subset(cvd_brazil, region == "SP" & date >= as.Date("2020-03-17") &
               date <= as.Date("2020-06-20"))
fit <- shewhart_regression(sp, value = new_deaths, index = date,
                           model = "log", limits_scale = "model",
                           phase_rule = "we_seven_same")
ggplot2::autoplot(fit, legend_position = "inside")
```

cvd_recife *COVID-19 daily mortality, Recife, Brazil, 2020*

Description

Daily count of new COVID-19 deaths officially recorded in Recife (capital of Pernambuco state, Brazil) between 28 March 2020 and 31 December 2020. Distributed with v0.1.x as `inst/extdata/recife_2020_covid19.r` and preserved in v1.0.0 as a vignette case study illustrating the regression-based control chart on non-stationary epidemiological counts.

Usage

```
cvd_recife
```

Format

A tibble with 279 rows and 3 columns:

date Date of the bulletin.

new_deaths Integer count of new deaths reported that day.

.t Integer row index (1..N), useful as a continuous predictor for `shewhart_regression()`.

Source

Castlab (Universidade Federal de Pernambuco) compiled the original series from the Brazilian Ministry of Health daily bulletins. See https://covid.saude.gov.br/.

See Also

`shewhart_regression()`, `vignette("covid-recife", package = "shewhartr")`.

Examples

```
fit <- shewhart_regression(cvd_recife,
                           value = new_deaths, index = .t,
                           model = "loglog")
ggplot2::autoplot(fit)
```

fit_gompertz_dummy *Convenience wrapper to fit* SSgompertzDummy *to a data frame*

Description

Convenience wrapper to fit SSgompertzDummy to a data frame

Usage

```
fit_gompertz_dummy(data, x, y, dummy, start = NULL, ...)
```

Arguments

data	A data frame with the columns referenced by x, y, dummy.
x, y, dummy	Tidy-eval column references.
start	Optional named list of starting values (Asym, b2, b3, Beta). If NULL, the self-starting initialiser is used.
...	Additional arguments passed to stats::nls().

Value

An object of class nls.

Examples

```
set.seed(42)
df <- data.frame(
  x = seq(1, 10, length.out = 50),
  d = rep(c(0, 1), each = 25)
)
df$y <- 100 * exp(-2 * exp(-0.3 * df$x)) + 20 * df$d + rnorm(50, 0, 3)
fit <- fit_gompertz_dummy(df, x = x, y = y, dummy = d)
coef(fit)
```

glance.shewhart_chart *Glance at a Shewhart chart's overall diagnostics*

Description

Returns a one-row tibble with overall chart-level diagnostics suitable for filling a row in a comparison table.

Usage

```
## S3 method for class 'shewhart_chart'
glance(x, ...)
```

Arguments

x	A shewhart_chart object.
...	Currently unused.

Value

A one-row tibble with columns type, n, phase, sigma_hat, sigma_method, n_violations, n_rules, pct_violations. n_violations counts rule hits (one row per point and rule), while pct_violations is the share of points flagged by at least one rule, so it always lies in [0, 1].

Examples

```
fit <- shewhart_i_mr(data.frame(y = rnorm(50)), value = y)
broom::glance(fit)
```

Gompertz *Gompertz growth function*

Description

Computes the value of the Gompertz curve parameterised in terms of starting value, asymptote, growth rate and lag:

$$G(x) = y_0 + (y_{\max} - y_0)\, \exp\left[-\exp\left(\frac{k\, e\, (\mathrm{lag} - x)}{y_{\max} - y_0} + 1\right)\right].$$

Usage

```
Gompertz(x, y0, ymax, k, lag)
```

Arguments

x	Numeric vector. The independent variable (e.g. time).
y0	Lower asymptote.
ymax	Upper asymptote.
k	Maximum specific growth rate.
lag	Lag time.

Details

This parameterisation, often called the "Zwietering Gompertz" form after Zwietering et al. (1990, eq. 3), gives directly interpretable parameters: `y0` is the lower asymptote, `ymax` the upper asymptote, `k` the maximum growth rate (the slope of the curve at its inflection point), and `lag` the lag time (where the tangent at the inflection point crosses `y0`). The factor e is what makes `k` the maximum slope; versions of shewhartr up to 1.3.0 omitted it, so their curves had maximum slope k/e.

Value

A numeric vector the same length as `x`.

References

Gompertz, B. (1825). On the Nature of the Function Expressive of the Law of Human Mortality. *Philosophical Transactions of the Royal Society of London*, 115, 513-583.

Zwietering, M. H., Jongenburger, I., Rombouts, F. M., & van 't Riet, K. (1990). Modeling of the Bacterial Growth Curve. *Applied and Environmental Microbiology*, 56(6), 1875-1881. doi:10.1128/aem.56.6.18751881.1990

See Also

`SSgompertzDummy()` for an `nls`-friendly self-starting variant that allows a covariate shift.

Examples

```
x <- seq(0, 30, by = 0.5)
y <- Gompertz(x, y0 = 0, ymax = 100, k = 5, lag = 5)
plot(x, y, type = "l", main = "Gompertz growth curve")
```

`iloglog` *Inverse log-log transformation*

Description

Inverts `loglog()`:

$$x = \alpha\,[\exp(\exp(y) - 1) - 1].$$

Usage

```
iloglog(x, alpha = 1)
```

Arguments

`x`	Numeric vector previously transformed via `loglog()`.
`alpha`	Same `alpha` used in `loglog()`. Default: 1.

Value

A numeric vector on the original scale.

Examples

```
original <- c(0, 1, 5, 10, 100, 1000)
all.equal(original, iloglog(loglog(original)))
```

`inv_box_cox` *Inverse Box-Cox transformation*

Description

Inverse Box-Cox transformation

Usage

```
inv_box_cox(x, lambda)
```

Arguments

`x`	Numeric vector of strictly positive values.
`lambda`	Numeric scalar. Power parameter.

Value

A numeric vector on the original scale.

Examples

```
y <- box_cox(1:10, lambda = 0.5)
inv_box_cox(y, lambda = 0.5)          # recovers 1:10
inv_box_cox(log(1:10), lambda = 0)    # lambda = 0 is exp()
```

is_shewhart_chart *Shewhart chart S3 class*

Description

All chart constructors in the package return an object of class `shewhart_chart` with a more specific subclass (`shewhart_i_mr`, `shewhart_xbar_r`, `shewhart_p`, `shewhart_regression`, ...). The shared slots are `type`, `augmented`, `limits`, `violations`, `fits`, `rules`, `sigma_hat`, `sigma_method`, `phase`, `n`, `call`, and `metadata`. `is_shewhart_chart()` tests inheritance.

Usage

```
is_shewhart_chart(x)
```

Arguments

`x`	An object.

Value

Logical scalar.

Examples

```
fit <- shewhart_i_mr(data.frame(v = rnorm(30)), v)
is_shewhart_chart(fit)
```

loglog *Log-log transformation*

Description

Applies the stabilising transformation

$$y = \log\bigl(\log\bigl(x/\alpha + 1\bigr) + 1\bigr).$$

Useful for very right-skewed non-negative data, particularly count processes with heavy tails. The +1 inside each log makes the transformation well-defined at zero.

Usage

```
loglog(x, alpha = 1)
```

Arguments

`x`	Numeric vector. Values must satisfy `x / alpha > -1`.
`alpha`	Positive scaling parameter. Default: 1.

Value

A numeric vector of transformed values.

See Also

`iloglog()` for the inverse, `shewhart_box_cox()` for a data-driven transformation choice.

Examples

```
x <- c(0, 1, 5, 10, 100, 1000)
loglog(x)
```

`monitor` *Phase II monitoring against pre-calibrated limits*

Description

Applies the control limits (and rule set) from a calibrated shewhart_chart object to fresh data. The new data must contain the same columns used by the original chart constructor.

Usage

```
monitor(data, chart)
```

Arguments

`data`	A data frame with the same columns as the data used to fit `chart`.
`chart`	A pre-calibrated shewhart_chart object.

Details

Limits are *not* re-estimated; they are the limits stored on the calibration object. Only the violation table is recomputed against the new observations.

Value

A shewhart_chart object with `$phase = "phase_2"` and limits inherited from `chart`.

References

Woodall, W. H. (2000). Controversies and Contradictions in Statistical Process Control. *Journal of Quality Technology*, 32(4), 341-350.

Examples

```
set.seed(1)
base    <- data.frame(y = rnorm(50, mean = 100, sd = 2))
new_obs <- data.frame(y = rnorm(20, mean = 102, sd = 2))   # small shift
calib   <- calibrate(base, value = y, chart = "i_mr")
alarms  <- monitor(new_obs, calib)
alarms$violations
```

---

pcb_solder *Solder defects on printed circuit boards*

Description

A synthetic Poisson dataset: number of defective solder joints on each of 50 inspected printed circuit boards. The mean is 6, ideal for either the 3-sigma c chart or its more honest cousin with exact Poisson limits.

Usage

```
pcb_solder
```

Format

A tibble with 50 rows and 2 columns:

board Integer board identifier.

defects Integer count of defective joints.

Source

Synthetic. See `data-raw/build_all.R`.

See Also

`shewhart_c()`.

Examples

```
fit       <- shewhart_c(pcb_solder, defects = defects, index = board)
fit_exact <- shewhart_c(pcb_solder, defects = defects, index = board,
                        limits = "poisson")
```

print.shewhart_chart *Print a Shewhart chart object*

Description

Concise summary including chart type, sample size, sigma estimate and any rule violations. For full per-row results, use augment(); for a tabular limit summary, use tidy().

Usage

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

Arguments

x	A shewhart_chart object.
...	Currently unused.

Value

Returns x invisibly (for chaining).

Examples

```
set.seed(1)
fit <- shewhart_i_mr(data.frame(y = rnorm(50)), value = y)
print(fit)
```

rolling_sum *Rolling sum with a configurable window*

Description

Slides over x summing the last .window elements (including the current one). Treats NA as zero. Used internally by the runs tests.

Usage

```
rolling_sum(x, .window = 7L)
```

Arguments

x	Numeric vector.
.window	Integer window size (>= 1). Default: 7.

Value

A numeric vector the same length as `x`.

Examples

```
rolling_sum(c(1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1), .window = 7)
```

---

shewhart_arl *Estimate Average Run Length via Monte Carlo simulation*

Description

Simulates the run length of a Shewhart-type chart configuration under a sequence of mean shifts. For each shift size, replicates draw normal observations with mean `shift * sigma` (relative to the in-control centre) and stop at the first observation where any of the supplied rules fires. Returns a tibble with the average run length and a Monte Carlo confidence interval.

Usage

```
shewhart_arl(
  shift = seq(0, 3, by = 0.5),
  rules = c("nelson_1_beyond_3s", "nelson_2_nine_same"),
  n_sim = 5000L,
  max_run = 1000L,
  sigma = 1,
  seed = NULL
)
```

Arguments

`shift`	Numeric vector of mean shifts to evaluate, in units of `sigma`. Default: `seq(0, 3, by = 0.5)`.
`rules`	Character vector of rule keys (see `shewhart_rules_available()`).
`n_sim`	Integer. Number of simulation replicates per shift. Default: 5000.
`max_run`	Integer. Maximum run length before truncation (a censored alarm). Default: 1000.
`sigma`	Numeric. In-control sigma. Default: 1 (the relevant quantity is `shift / sigma`, so 1 is fine).
`seed`	Optional integer seed for reproducibility.

Details

Computational note: for `n_sim = 1e4` and a moderate set of rules, a single configuration takes a fraction of a second; a fine shift grid (10-15 points) takes a few seconds. For tighter intervals or larger rule sets, increase `n_sim` and/or set `parallel = TRUE` (currently a placeholder for future implementation).

Value

A tibble with columns `shift`, `arl`, `arl_se`, `arl_lower`, `arl_upper`, `n_truncated`. `n_truncated` counts how many replicates hit `max_run` before alarming (a sign that `max_run` should be raised).

References

Champ, C. W., & Woodall, W. H. (1987). Exact Results for Shewhart Control Charts with Supplementary Runs Rules. *Technometrics*, 29(4), 393-399. doi:10.1080/00401706.1987.10488262

Wald, A. (1947). *Sequential Analysis*. Wiley.

Examples

```
# In-control ARL of Nelson 1 (closed-form ARL_0 ~= 370.4)
set.seed(1)
shewhart_arl(shift = 0, rules = "nelson_1_beyond_3s",
             n_sim = 2000, max_run = 2000)

# Adding Nelson 2 sharpens detection but lowers ARL_0
shewhart_arl(shift = c(0, 0.5, 1, 1.5, 2),
             rules = c("nelson_1_beyond_3s", "nelson_2_nine_same"),
             n_sim = 2000)
```

---

shewhart_box_cox *Box-Cox profile log-likelihood*

Description

Computes the profile log-likelihood for the Box-Cox power parameter lambda, applied to a single positive numeric series (regression to a constant) or to the residuals of a linear model.

Usage

```
shewhart_box_cox(data, value = NULL, lambda_grid = seq(-2, 2, by = 0.05))
```

Arguments

`data`	A data frame, or a numeric vector. If a data frame, `value` must be supplied.
`value`	Tidy-eval column reference for the response (only when `data` is a data frame).
`lambda_grid`	Numeric vector of lambda values to evaluate. Default: a fine grid from -2 to 2.

Details

Reports the lambda that maximises the profile likelihood and a 95% confidence interval based on the chi-square approximation to twice the log-likelihood drop (Box & Cox 1964, eq. 9).

Value

An object of class `shewhart_box_cox` with components `profile` (tibble of lambda vs. log-likelihood), `lambda_hat` (the maximiser), `ci` (95% CI). The object has its own `print()` method.

References

Box, G. E. P., & Cox, D. R. (1964). An Analysis of Transformations. *Journal of the Royal Statistical Society, Series B*, 26(2), 211-252. doi:10.1111/j.25176161.1964.tb00553.x

Examples

```
set.seed(1)
bc <- shewhart_box_cox(rlnorm(200, meanlog = 0, sdlog = 0.5))
bc$lambda_hat   # should be near 0 (log-normal data)
```

`shewhart_c` *c chart for the number of nonconformities*

Description

Constructs a c chart for counts of nonconformities (defects) per inspection unit, where the unit (area, length, time, etc.) is constant across observations. For variable inspection size use `shewhart_u()`.

Usage

```
shewhart_c(
  data,
  defects,
  index = NULL,
  limits = c("3sigma", "poisson"),
  rules = c("nelson_1_beyond_3s"),
  locale = getOption("shewhart.locale", "en"),
  verbose = NULL
)
```

Arguments

`data`	A data frame.
`defects`	Tidy-eval column reference for the count of nonconformities per inspection unit.
`index`	Optional tidy-eval column reference for the x-axis.
`limits`	One of `"3sigma"` (default; classical normal approximation) or `"poisson"` (exact Poisson 0.00135 / 0.99865 quantiles, recommended when `c_bar < 10`).
`rules`	Character vector of rule keys to apply. See `shewhart_rules_available()`. Default applies Nelson 1 and 2.
`locale`	One of `"en"`, `"pt"`, `"es"`, `"fr"`. Affects plot labels and informative messages.
`verbose`	Logical. Print progress messages? Defaults to the `shewhart.verbose` option.

Value

A shewhart_chart object of subclass `shewhart_c`.

References

Montgomery, D. C. (2019). *Introduction to Statistical Quality Control* (8th ed.). Wiley. Chapter 7.3.

Ryan, T. P. (2011). *Statistical Methods for Quality Improvement* (3rd ed.). Wiley. Chapter 6 (on the inadequacy of 3-sigma limits for low-mean Poisson counts).

Examples

```
set.seed(1)
df <- data.frame(
  unit    = 1:40,
  defects = rpois(40, lambda = 6)
)
fit <- shewhart_c(df, defects = defects, index = unit)
fit_exact <- shewhart_c(df, defects = defects, index = unit,
                        limits = "poisson")
```

`shewhart_capability` *Process capability indices Cp, Cpk, Pp, Ppk*

Description

Computes the four classical capability indices for a Shewhart chart or a raw vector. Optionally returns bootstrap confidence intervals.

Usage

```
shewhart_capability(
  data,
  lsl = NA_real_,
  usl = NA_real_,
  target = NA_real_,
  ci_level = 0.95,
  n_boot = 2000L,
  seed = NULL
)
```

Arguments

`data`	A shewhart_chart object or a numeric vector.
`lsl, usl`	Numeric scalars. Lower and upper specification limits. At least one must be supplied.

target	Numeric scalar. Optional process target. If missing, defaults to the midpoint of (lsl, usl).
ci_level	Numeric. Confidence level for bootstrap intervals. Default 0.95. Set to NA to skip bootstrap.
n_boot	Integer. Number of bootstrap replicates. Default 2000.
seed	Optional integer for reproducibility.

Details

For a shewhart_chart of type i_mr, ewma, cusum, xbar_r or xbar_s, the within sigma stored on the chart object (sigma_hat, already expressed per individual measurement) is used for Cp/Cpk; the overall standard deviation of the individual measurements is used for Pp/Ppk. For subgroup charts the individual measurements are the ones the chart was built from (not the subgroup means). For a numeric vector data, a single sigma is used for both pairs (so Cp = Pp and Cpk = Ppk). Other chart types (attribute, regression, multivariate) are not supported; pass the measurements as a numeric vector instead.

Bootstrap intervals re-estimate sigma within in every replicate with the same estimator as the chart: whole subgroups are resampled for Xbar-R / Xbar-S charts, and consecutive pairs of observations (so that each replicate keeps its moving ranges) for individual charts.

Capability indices are only meaningful when the process is in statistical control (Phase I). The function emits a warning if the supplied chart has any rule violations.

Value

A list of class shewhart_capability with point estimates and (optionally) bootstrap CIs.

References

Kotz, S., & Lovelace, C. R. (1998). *Process Capability Indices in Theory and Practice*. Arnold.

Montgomery, D. C. (2019). *Introduction to Statistical Quality Control* (8th ed.). Wiley. Chapter 8.

Pearn, W. L., & Kotz, S. (2006). *Encyclopedia and Handbook of Process Capability Indices*. World Scientific.

Examples

```
set.seed(1)
df <- data.frame(y = rnorm(100, mean = 50, sd = 0.8))
fit <- shewhart_i_mr(df, value = y)
cap <- shewhart_capability(fit, lsl = 47, usl = 53, target = 50)
print(cap)
```

shewhart_constants *Look up Shewhart control chart constants*

Description

Returns the classical Shewhart constants (A2, A3, c4, d2, d3, B3-B6, D3-D4) for a given subgroup size `n`. Tabulated values are used for `n <= 25`; for larger samples, c4 is computed from its closed form and the remaining constants are derived analytically when known (otherwise NA is returned with a warning).

Usage

```
shewhart_constants(n)
```

Arguments

`n`	Integer scalar or vector. Subgroup size (>= 2).

Value

A data frame with columns `n`, `A2`, `A3`, `c4`, `d2`, `d3`, `B3`, `B4`, `B5`, `B6`, `D3`, `D4`.

References

Montgomery, D. C. (2019). *Introduction to Statistical Quality Control* (8th ed.). Wiley. Appendix VI.

Examples

```
shewhart_constants(5)
shewhart_constants(c(2, 5, 10, 25))
```

shewhart_cusum *Tabular CUSUM control chart*

Description

Constructs a two-sided tabular CUSUM chart for a single column of individual measurements. Two cumulative statistics, `C+` (upward) and `C-` (downward), are accumulated against a target with a reference value `k`; an alarm fires when either crosses the decision interval `h * sigma`.

Usage

```
shewhart_cusum(
  data,
  value,
  index = NULL,
  target = NULL,
  sigma = NULL,
  k = 0.5,
  h = 4,
  locale = getOption("shewhart.locale", "en"),
  verbose = NULL
)
```

Arguments

data	A data frame.
value	Tidy-eval column reference for the measurement.
index	Optional tidy-eval column reference for the x-axis.
target	Numeric. Process target. Defaults to mean(value).
sigma	Numeric. Process sigma. Defaults to MR_bar / 1.128.
k	Numeric. Reference value in units of sigma. Default 0.5, tuned to detect 1-sigma shifts.
h	Numeric. Decision interval in units of sigma. Default 4, giving ARL_0 ~ 168 for k = 0.5. Use h = 5 for ARL_0 ~ 465 (Hawkins & Olwell 1998).
locale	One of "en", "pt", "es", "fr".
verbose	Logical. Print progress messages?

Details

By default, sigma is estimated from the moving range of value (MR_bar / 1.128); the target is the mean of value. Either can be overridden via target and sigma for Phase II monitoring against pre-calibrated values. Missing values in value are an error, since a single NA would propagate through the accumulators and silently disable every later alarm.

Value

A shewhart_chart object of subclass shewhart_cusum. The augmented slot has columns .value, .cusum_pos, .cusum_neg (the two accumulated statistics, both non-negative), .upper (the decision interval h * sigma), and .flag_signal.

References

Page, E. S. (1954). Continuous Inspection Schemes. *Biometrika*, 41(1-2), 100-115. doi:10.1093/biomet/41.12.100

Hawkins, D. M., & Olwell, D. H. (1998). *Cumulative Sum Charts and Charting for Quality Improvement*. Springer.

Montgomery, D. C. (2019). *Introduction to Statistical Quality Control* (8th ed.). Wiley. Chapter 9.

Examples

```
set.seed(1)
df <- data.frame(
  day = 1:80,
  y   = c(rnorm(40, mean = 100, sd = 2),
          rnorm(40, mean = 101, sd = 2))   # 0.5 sigma shift
)
fit <- shewhart_cusum(df, value = y, index = day)
print(fit)

ggplot2::autoplot(fit)
```

`shewhart_diagnostics` *Tukey-style residual diagnostic panel*

Description

For chart objects whose residuals are meaningful (`shewhart_i_mr`, `shewhart_xbar_r`, `shewhart_xbar_s`, `shewhart_regression`), produces the five-panel residual diagnostic favoured by exploratory data analysis: residuals vs. fitted, normal Q-Q, autocorrelation, moving-range plot of residuals, residual histogram. The aim is to make the assumptions that the chart is making visible: independence (ACF), normality (Q-Q, histogram), constant variance (residuals vs. fitted), and the absence of trend in dispersion (moving range).

Usage

```
shewhart_diagnostics(chart, locale = NULL)
```

Arguments

`chart`	A shewhart_chart object.
`locale`	Optional override for the chart's stored locale.

Value

A list of `ggplot` objects with class `shewhart_diagnostics`. The print method composes the panels.

References

Tukey, J. W. (1977). *Exploratory Data Analysis*. Addison-Wesley.

Box, G. E. P., Hunter, W. G., & Hunter, J. S. (2005). *Statistics for Experimenters: Design, Innovation, and Discovery* (2nd ed.). Wiley.

Examples

```
fit <- shewhart_i_mr(data.frame(y = rnorm(100)), value = y)
print(shewhart_diagnostics(fit))
```

---

shewhart_ewma *Exponentially Weighted Moving Average (EWMA) control chart*

Description

Constructs an EWMA chart for a single column of individual measurements. The chart is more sensitive than a Shewhart I chart to small but persistent shifts in the process mean, at the cost of a longer reaction time to large shifts.

Usage

```
shewhart_ewma(
  data,
  value,
  index = NULL,
  target = NULL,
  sigma = NULL,
  lambda = 0.2,
  L = 2.86,
  steady_state = FALSE,
  rules = "nelson_1_beyond_3s",
  locale = getOption("shewhart.locale", "en"),
  verbose = NULL
)
```

Arguments

data	A data frame.
value	Tidy-eval column reference for the measurement.
index	Optional tidy-eval column reference for the x-axis.
target	Numeric. Process target / centre line. Defaults to mean(value).
sigma	Numeric. Process sigma. Defaults to MR_bar / 1.128.
lambda	Numeric in (0, 1]. Smoothing constant. Default 0.2. Smaller lambda = more memory, more sensitive to small shifts.
L	Numeric. Width of the limits in standard errors of the EWMA. Default 2.86, which combined with lambda = 0.2 yields ARL_0 ~ 370 (Lucas & Saccucci 1990).
steady_state	Logical. Use asymptotic (constant) limits instead of time-varying ones?

rules	Character vector of runs rules to flag. Defaults to Nelson 1 only, which fires exactly when .ewma crosses .upper/.lower. The rules are evaluated on the EWMA series with a sigma-equivalent of L * se / 3, so the zones used by rules 5-8 are thirds of the distance from the centre to the EWMA limit (they equal 1 and 2 standard errors only when L = 3). The higher-order Nelson rules are not designed for autocorrelated statistics and their false-alarm rates on the EWMA are not calibrated.
locale	One of "en", "pt", "es", "fr".
verbose	Logical. Print progress messages?

Details

By default, sigma is estimated from the moving range of value (Wheeler 1992 convention, MR_bar / 1.128); the centre is the mean of value. Either can be overridden via target and sigma for Phase II monitoring against pre-calibrated values. Missing values in value are an error, since a single NA would propagate through the recursion and silently disable every later alarm.

Limits are time-varying by default: they widen out from target as the EWMA "warms up", converging to the asymptotic limits as i -> infinity. Set steady_state = TRUE to use the asymptotic limits everywhere (commonly chosen when calibrating from a long baseline).

Value

A shewhart_chart object of subclass shewhart_ewma. The augmented slot has columns .value (the original observation), .ewma (the smoothed statistic z_i, plotted on the chart), and the usual .center, .upper, .lower, .flag_*.

References

Roberts, S. W. (1959). Control Chart Tests Based on Geometric Moving Averages. *Technometrics*, 1(3), 239-250. doi:10.1080/00401706.1959.10489860

Lucas, J. M., & Saccucci, M. S. (1990). Exponentially Weighted Moving Average Control Schemes: Properties and Enhancements. *Technometrics*, 32(1), 1-12. doi:10.1080/00401706.1990.10484583

Montgomery, D. C. (2019). *Introduction to Statistical Quality Control* (8th ed.). Wiley. Chapter 9.

Examples

```
set.seed(1)
df <- data.frame(
  day = 1:80,
  y   = c(rnorm(40, mean = 100, sd = 2),
          rnorm(40, mean = 101, sd = 2))   # 0.5 sigma shift
)
fit <- shewhart_ewma(df, value = y, index = day)
print(fit)

ggplot2::autoplot(fit)
```

shewhart_hotelling *Hotelling T-squared multivariate control chart*

Description

Constructs a Hotelling T^2 chart for joint monitoring of p correlated quality characteristics. Use this chart when the variables genuinely co-vary — a classical example is a chemical process where temperature, pressure and flow rate are mechanically coupled, and a fault that breaks the coupling moves them off the joint distribution but possibly stays inside each marginal limit.

Usage

```
shewhart_hotelling(
  data,
  vars,
  subgroup = NULL,
  index = NULL,
  phase = c("phase_1", "phase_2"),
  alpha = 0.0027,
  locale = getOption("shewhart.locale", "en"),
  verbose = NULL
)
```

Arguments

data	A data frame.
vars	Tidy-select expression for the columns containing the variables to monitor jointly (c(x1, x2, x3), tidyselect::starts_with("temp"), etc.). Must select at least 2 columns.
subgroup	Optional tidy-eval column for rational subgrouping. If supplied, all rows sharing a value of this column are treated as a single subgroup. If NULL (default), every row is its own observation (individual-observations chart). Subgroups are kept in order of first appearance in data (time order).
index	Optional tidy-eval column for the x-axis. If supplied, must vary across observations (or across subgroups, if subgroup is supplied).
phase	One of "phase_1" (default; retrospective) or "phase_2" (prospective monitoring of new observations against parameters estimated from the same data).
alpha	Type-I error rate per observation. Default 0.0027, matching the conventional Shewhart 3-sigma false-alarm rate.
locale	One of "en", "pt", "es", "fr".
verbose	Logical. Print progress messages?

Details

Both individual observations (`subgroup = NULL`) and rationally subgrouped observations (`subgroup` supplied) are supported. The chart selects the appropriate exact small-sample limits for the selected `phase` (Phase I uses retrospective limits derived from a Beta or F distribution; Phase II uses the slightly wider limits that propagate the Phase I parameter uncertainty to a fresh observation).

Value

A shewhart_chart object of subclass `shewhart_hotelling`. The augmented tibble has columns `.t2` (the statistic), `.upper` (UCL — constant within a chart), `.flag_signal` and `.flag_any`, and one `.contrib_<var>` column per monitored variable giving that variable's marginal contribution to the alarm (Mason et al. 1995). The `limits` slot contains the chart-level UCL; the `metadata` slot stores the variable names, subgroup column name, and the parameters `p`, `m`, `n`, `phase`, `alpha` that determined the limit.

References

Hotelling, H. (1947). Multivariate quality control. In: *Techniques of Statistical Analysis*. McGraw-Hill.

Tracy, N. D., Young, J. C., & Mason, R. L. (1992). Multivariate control charts for individual observations. *Journal of Quality Technology*, 24(2), 88-95. doi:10.1080/00224065.1992.11979383

Mason, R. L., Tracy, N. D., & Young, J. C. (1995). Decomposition of T^2 for multivariate control chart interpretation. *Journal of Quality Technology*, 27(2), 99-108. doi:10.1080/00224065.1995.11979573

Mason, R. L., & Young, J. C. (2002). *Multivariate Statistical Process Control with Industrial Applications*. SIAM/ASA.

Montgomery, D. C. (2019). *Introduction to Statistical Quality Control* (8th ed.). Wiley. Chapter 11.

Examples

```
set.seed(1)
Sigma <- matrix(c(1, 0.7, 0.7, 1), 2, 2)
Z     <- MASS::mvrnorm(60, c(0, 0), Sigma)
df    <- tibble::tibble(t = 1:60, x1 = Z[, 1], x2 = Z[, 2])
fit   <- shewhart_hotelling(df, vars = c(x1, x2), index = t)
print(fit)

ggplot2::autoplot(fit)
```

---

shewhart_i_mr *Individuals and Moving Range (I-MR) control chart*

Description

Constructs an I-MR chart for a single column of individual measurements. Returns a shewhart_chart object that supports `print()`, `summary()`, `autoplot()`, `tidy()`, `glance()` and `augment()`.

Usage

```
shewhart_i_mr(
  data,
  value,
  index = NULL,
  sigma_method = c("mr", "median_mr", "biweight", "sd"),
  rules = c("nelson_1_beyond_3s", "nelson_2_nine_same"),
  locale = getOption("shewhart.locale", "en"),
  verbose = NULL
)
```

Arguments

data	A data frame.
value	Tidy-eval column reference for the measurement.
index	Optional tidy-eval column reference for the x-axis (date or integer). If NULL (default), the row index is used.
sigma_method	One of "mr" (default; classical moving range), "median_mr" (Tukey-style robust: median of moving ranges, with bias correction), "biweight" (Tukey biweight midvariance), "sd" (sample SD).
rules	Character vector of rule keys to apply. See shewhart_rules_available(). Default applies Nelson 1 and 2.
locale	One of "en", "pt", "es", "fr". Affects plot labels and informative messages.
verbose	Logical. Print progress messages? Defaults to the shewhart.verbose option.

Details

Sigma is estimated from the moving range with d2(2) = 1.128; the classical 3-sigma limits are equivalent to x_bar +/- 2.660 * MR_bar. The MR chart limits are [0, D4(2) * MR_bar] with D4(2) = 3.267.

Value

A shewhart_chart object of subclass shewhart_i_mr.

References

Montgomery, D. C. (2019). *Introduction to Statistical Quality Control* (8th ed.). Wiley. Chapter 6.

Wheeler, D. J., & Chambers, D. S. (1992). *Understanding Statistical Process Control* (2nd ed.). SPC Press.

Examples

```
set.seed(1)
df <- data.frame(
  day = seq.Date(as.Date("2024-01-01"), by = "day", length.out = 60),
  y   = c(rnorm(40, mean = 100, sd = 2),
```

```
          rnorm(20, mean = 103, sd = 2))   # shift after position 40
)
fit <- shewhart_i_mr(df, value = y, index = day)
print(fit)

ggplot2::autoplot(fit)
```

---

shewhart_mcusum *Multivariate CUSUM control chart (Crosier 1988)*

Description

Constructs a multivariate CUSUM chart for jointly monitoring p correlated variables. Like the univariate CUSUM it accumulates deviations from a target with a reference value k that decides when the accumulator resets; unlike a Hotelling T^2 chart it carries memory across observations and so detects small persistent shifts faster.

Usage

```
shewhart_mcusum(
  data,
  vars,
  index = NULL,
  target = NULL,
  cov = NULL,
  k = 0.5,
  h = NULL,
  locale = getOption("shewhart.locale", "en"),
  verbose = NULL
)
```

Arguments

data	A data frame.
vars	Tidy-select expression for the columns to monitor jointly. At least 2 columns.
index	Optional tidy-eval column for the x-axis.
target	Optional length-p numeric vector. The in-control mean. Defaults to colMeans(data[, vars]).
cov	Optional p x p covariance matrix. Defaults to cov(data[, vars]).
k	Reference value, in sigma units. Default 0.5, tuned for shifts of 1 sigma. Lower k makes the chart sensitive to smaller shifts but increases false alarms.
h	Decision interval. If NULL, looked up in a table giving ARL_0 ~ 200 for k = 0.5 and p = 2..10 (simulated with known in-control parameters; p = 2 matches Crosier 1988). With target and cov estimated from the same data the actual in-control ARL differs; pass h explicitly for other designs.

`locale`	One of `"en"`, `"pt"`, `"es"`, `"fr"`.
`verbose`	Logical. Print progress messages?

Value

A shewhart_chart object of subclass `shewhart_mcusum`. The augmented tibble has columns `.y` (the chart statistic), `.upper` (the decision interval `h`), and `.flag_signal`.

References

Crosier, R. B. (1988). Multivariate Generalizations of Cumulative Sum Quality-Control Schemes. *Technometrics*, 30(3), 291-303. doi:10.1080/00401706.1988.10488402

Pignatiello, J. J., & Runger, G. C. (1990). Comparisons of Multivariate CUSUM Charts. *Journal of Quality Technology*, 22(3), 173-186. doi:10.1080/00224065.1990.11979237

Examples

```
set.seed(1)
Sigma <- matrix(c(1, 0.6, 0.6, 1), 2, 2)
base  <- MASS::mvrnorm(60, c(0, 0), Sigma)
shift <- MASS::mvrnorm(40, c(0.6, 0.6), Sigma)
df    <- data.frame(t = 1:100,
                    x1 = c(base[, 1], shift[, 1]),
                    x2 = c(base[, 2], shift[, 2]))
fit <- shewhart_mcusum(df, vars = c(x1, x2), index = t,
                       target = c(0, 0), cov = Sigma)
print(fit)

ggplot2::autoplot(fit)
```

`shewhart_mewma` *Multivariate EWMA control chart*

Description

Constructs a multivariate Exponentially Weighted Moving Average (MEWMA) chart for jointly monitoring `p` correlated variables. The chart is more sensitive than the Hotelling `T^2` chart to small persistent shifts in the *vector* mean, in the same way the univariate EWMA is more sensitive than a Shewhart I chart.

Usage

```
shewhart_mewma(
  data,
  vars,
  index = NULL,
  target = NULL,
```

```
  cov = NULL,
  lambda = 0.1,
  h = NULL,
  steady_state = FALSE,
  locale = getOption("shewhart.locale", "en"),
  verbose = NULL
)
```

Arguments

data	A data frame.
vars	Tidy-select expression for the columns to monitor jointly. Must select at least 2 columns.
index	Optional tidy-eval column for the x-axis.
target	Optional length-p numeric vector. The in-control mean. Defaults to `colMeans(data[, vars])`.
cov	Optional `p x p` covariance matrix. Defaults to `cov(data[, vars])`.
lambda	Smoothing constant in `(0, 1]`. Default `0.1`.
h	Decision interval (UCL on the `T^2` statistic). If `NULL`, looked up in the `ARL_0 ~ 200` table matching `steady_state`.
steady_state	Logical. Use the steady-state covariance `(lambda / (2 - lambda)) * Sigma` everywhere instead of the time-varying form? Default `FALSE`.
locale	One of `"en"`, `"pt"`, `"es"`, `"fr"`.
verbose	Logical. Print progress messages?

Details

By default `target` (the in-control mean vector) and `cov` (the in-control covariance) are estimated from the data. For Phase II monitoring, supply both explicitly so the limits use the calibration values. The decision interval `h` is calibrated by lookup in a table giving `ARL_0 ~ 200` (with known in-control parameters) for `lambda` in `{0.05, 0.10, 0.20, 0.40}` and `p = 2..6`. There is one table for each covariance mode and the lookup follows `steady_state`; the steady-state table reproduces Prabhu & Runger (1997). If the `(lambda, p)` combination is outside the tabulated range, the user must pass `h` explicitly.

In Phase II (`monitor()`), the recursion continues from the last Phase I vector `Z` (and from the previous batch when `monitor()` is chained), with the covariance of the continued recursion.

Value

A shewhart_chart object of subclass `shewhart_mewma`. The augmented tibble has columns `.t2` (the MEWMA statistic), `.upper` (the decision interval `h`), and `.flag_signal`.

References

Lowry, C. A., Woodall, W. H., Champ, C. W., & Rigdon, S. E. (1992). A Multivariate Exponentially Weighted Moving Average Control Chart. *Technometrics*, 34(1), 46-53. doi:10.1080/00401706.1992.10485232

Prabhu, S. S., & Runger, G. C. (1997). Designing a Multivariate EWMA Control Chart. *Journal of Quality Technology*, 29(1), 8-15. doi:10.1080/00224065.1997.11979721

Examples

```
set.seed(1)
Sigma <- matrix(c(1, 0.6, 0.6, 1), 2, 2)
base  <- MASS::mvrnorm(60, c(0, 0), Sigma)
shift <- MASS::mvrnorm(40, c(0.4, 0.4), Sigma)         # 0.4 sigma shift
df    <- data.frame(t = 1:100,
                    x1 = c(base[, 1], shift[, 1]),
                    x2 = c(base[, 2], shift[, 2]))
fit <- shewhart_mewma(df, vars = c(x1, x2), index = t,
                      target = c(0, 0), cov = Sigma,
                      lambda = 0.1)
print(fit)

ggplot2::autoplot(fit)
```

---

shewhart_np *np chart for the number of nonconforming items*

Description

Constructs an np chart from counts of nonconforming items in subgroups of constant size `n`. For variable subgroup sizes, use `shewhart_p()` instead.

Usage

```
shewhart_np(
  data,
  defects,
  n,
  index = NULL,
  rules = c("nelson_1_beyond_3s"),
  locale = getOption("shewhart.locale", "en"),
  verbose = NULL
)
```

Arguments

`data`	A data frame.
`defects`	Tidy-eval column reference for the count of nonconforming items in each subgroup.
`n`	Numeric scalar. The (constant) subgroup size.
`index`	Optional tidy-eval column reference for the x-axis.

rules	Character vector of rule keys to apply. See shewhart_rules_available(). Default applies Nelson 1 and 2.
locale	One of "en", "pt", "es", "fr". Affects plot labels and informative messages.
verbose	Logical. Print progress messages? Defaults to the shewhart.verbose option.

Value

A shewhart_chart object of subclass shewhart_np.

References

Montgomery, D. C. (2019). *Introduction to Statistical Quality Control* (8th ed.). Wiley. Chapter 7.2.2.

Examples

```
set.seed(1)
df <- data.frame(
  day     = 1:30,
  defects = rbinom(30, size = 200, prob = 0.04)
)
fit <- shewhart_np(df, defects = defects, n = 200, index = day)
print(fit)
```

---

shewhart_p *p chart for the proportion of nonconforming items*

Description

Constructs a p chart from counts of nonconforming items in subgroups of size n. Subgroup sizes may vary; in that case, control limits are computed per observation.

Usage

```
shewhart_p(
  data,
  defects,
  n,
  index = NULL,
  limits = c("3sigma", "binomial"),
  rules = c("nelson_1_beyond_3s"),
  locale = getOption("shewhart.locale", "en"),
  verbose = NULL
)
```

Arguments

data	A data frame.
defects	Tidy-eval column reference for the count of nonconforming items in each subgroup.
n	Tidy-eval column reference for the subgroup size.
index	Optional tidy-eval column reference for the x-axis.
limits	One of "3sigma" (default; classical normal approximation) or "binomial" (exact 0.00135 / 0.99865 binomial quantiles, equivalent to the standard 3-sigma rate when n is large).
rules	Character vector of rule keys to apply. See shewhart_rules_available(). Default applies Nelson 1 and 2.
locale	One of "en", "pt", "es", "fr". Affects plot labels and informative messages.
verbose	Logical. Print progress messages? Defaults to the shewhart.verbose option.

Details

Standard 3-sigma limits use the normal approximation to the binomial:

$$\bar{p} \pm 3\sqrt{\bar{p}(1-\bar{p})/n_i}.$$

For very small n_i or very small / very large p_bar, the approximation deteriorates and exact binomial limits should be preferred (limits = "binomial").

Value

A shewhart_chart object of subclass shewhart_p.

References

Montgomery, D. C. (2019). *Introduction to Statistical Quality Control* (8th ed.). Wiley. Chapter 7.

Examples

```
set.seed(1)
df <- data.frame(
  day     = 1:30,
  defects = rbinom(30, size = 100, prob = 0.05),
  n       = 100
)
fit <- shewhart_p(df, defects = defects, n = n, index = day)
print(fit)
```

shewhart_palette *Access the package's named colour palettes*

Description

Returns one of the curated colour vectors that every chart in the package draws from. Useful when extending an `autoplot()` chart with your own layers and you want them to match the rest of the package's visual identity.

Usage

```
shewhart_palette(
  name = c("phase_seq", "family", "signal", "neutral"),
  n = NULL
)
```

Arguments

`name`	One of `"phase_seq"`, `"family"`, `"signal"`, `"neutral"`. See *Details* for the contents and intended use of each.
`n`	Optional integer. Number of colours to return. If `NULL` (default) the full palette is returned. If `n` exceeds the palette length, a smooth ramp (`grDevices::colorRampPalette()`) is interpolated.

Details

`phase_seq` Sequential palette for time-ordered phase indices in regression / multi-phase charts. Cool blues at the baseline, warming through neutral to terracotta as phase index grows. Avoids saturated red (reserved for `signal`).

`family` Named categorical palette for the four chart families: variables, attributes, memory-based, multivariate. Identical to the colours used in the architecture diagram.

`signal` Two-colour palette: `in_control` (deep blue) and `out_of_control` (firebrick). Used to colour violation points on every chart so the alarm signal is consistent.

`neutral` Structural greys used by `shewhart_theme()` for backgrounds, gridlines and text colour.

Value

A character vector of colour hex codes; for `family` and `signal`, the vector is named.

Examples

```
shewhart_palette("phase_seq", n = 4)
shewhart_palette("family")["multivariate"]
shewhart_palette("signal")
```

shewhart_regression *Regression-based control chart for processes with trend*

Description

Fits a chosen model to the data (linear, log, log-log, Gompertz, logistic, or a user-supplied formula), then constructs control limits around the fitted curve using the moving-range estimator on the residuals (Wheeler 1992). Optionally detects phase changes automatically via runs tests on the residuals and re-fits each phase.

Usage

```
shewhart_regression(
  data,
  value,
  index,
  model = c("auto", "linear", "log", "loglog", "gompertz", "logistic"),
  formula = NULL,
  dummy = NULL,
  start_base = NULL,
  phase_changes = NULL,
  phase_rule = "nelson_2_nine_same",
  rules = c("nelson_1_beyond_3s", "nelson_2_nine_same"),
  sigma_method = c("mr", "median_mr", "biweight", "sd"),
  lower_bound = NA_real_,
  limits_scale = c("original", "model"),
  locale = getOption("shewhart.locale", "en"),
  verbose = NULL
)
```

Arguments

data	A data frame.
value	Tidy-eval column reference for the response.
index	Tidy-eval column reference for the predictor (typically time, but can be any continuous variable).
model	Character. One of "auto" (Box-Cox guidance), "linear", "log" (fits log(y + 1) ~ N), "loglog", "gompertz", "logistic". For full control, supply formula instead. With "auto", each phase gets the Box-Cox profile maximiser lambda of y + 1, rounded to the nearest rung of the ladder that the menu offers: lambda < 0.25 selects "log", any larger value selects "linear" (a square-root lambda near 0.5 has no dedicated model and maps to "linear"; "loglog" is a stronger transform than log and is never chosen automatically).
formula	Optional one-sided or two-sided formula referencing columns in data. If provided, overrides model. The model may reference .N, the position of the observation within its phase. When the left-hand side is a recognised transformation of the response – log(y), log(y + c), log(y, base), log10(), log2(),

log1p() or sqrt(), optionally wrapped in I() – the fitted values are back-transformed to the scale of the data and limits_scale = "model" is available.

dummy — Optional tidy-eval column reference for an additive covariate (a "dummy" in the original v0.1 nomenclature; can be any factor or numeric covariate the user wants to adjust for, such as day-of-week effects or treatment indicators).

start_base — Integer or NULL. Number of initial observations used to estimate the first (base) phase. With automatic phase detection, NULL (the default) means 10. When phase_changes is supplied, NULL means the base phase ends just before the first supplied change; an explicit start_base adds a cut at observation start_base + 1 in addition to phase_changes.

phase_changes — Optional vector of index values at which to force a phase change (the observation whose index equals the value starts the new phase). If NULL, phase changes are detected automatically using the supplied phase_rule. A zero-length vector (e.g. integer(0)) fits a single phase without detection; it then needs only 3 observations, which is what a prospective replay needs to calibrate one phase at a time (see Details).

phase_rule — Character. Runs rule used to detect new phases. See shewhart_rules_available(). Default Nelson 2 (9 points same side; ARL_0 = 2^9 - 1 = 511). For backward compatibility with v0.1.x, use "we_seven_same" (7 points; ARL_0 = 2^7 - 1 = 127).

rules — Character vector of rules to flag on the final chart.

sigma_method — One of "mr" (default), "median_mr", "biweight" (Tukey-style robust), or "sd".

lower_bound — Numeric scalar or NA. If non-NA, lower limit is clipped at this value (commonly 0 for counts). Default NA (no clipping). Stored in metadata and honoured by monitor().

limits_scale — Scale on which the limits are built. With "original" (default), sigma is estimated from the residuals on the scale of the data and the band fitted +- 3 sigma is symmetric. With "model", sigma is estimated from the moving ranges of the residuals on the transformed scale of the model's left-hand side g(y); the band g_hat +- 3 sigma is formed there and centre line and limits are back-transformed with g^-1. With model = "log" this gives the multiplicative, asymmetric bands of Perla et al. (2020) and Ferraz et al. (2020) – an individuals chart on the residuals of log(1 + y) ~ t – whose lower limit never falls below 0. For models whose left-hand side is the raw response ("linear", "gompertz", "logistic", or a formula such as y ~ .N) the two options coincide. With "model", the .sigma column of the augmented tibble (and metadata$phase_sigma, sigma_hat) is on the model scale, and two extra columns hold what the runs rules are applied to: .model_value (the transformed observation g(y)) and .model_center (the fitted value g_hat).

locale — Character. One of "en", "pt", "es", "fr".

verbose — Logical. Print progress messages?

Details

This is the package's flagship chart, intended for trended or non-stationary processes for which classical Shewhart charts give systematically wrong limits. See the vignette regression-charts for a thorough discussion and examples.

Models that take the logarithm of the response (`"log"`, `"loglog"`, and `"auto"` whenever it selects `"log"`) require non-negative values: `log(1 + y)` is undefined for a negative count such as a bulletin that revises a cumulative total downwards. The function stops with an error naming the offending row instead of returning `NaN` limits; filter or reconcile such rows first.

A Phase I chart refits every phase, including the last one (labelled "Monitoring"). To judge new observations against the limits of the last phase *projected forward*, as in a prospective analysis, calibrate on the data up to the end of the last phase with `calibrate()` and pass the rest to `monitor()`: it continues `.N`, uses the stored sigma, and honours `limits_scale` and `lower_bound`. A prospective replay in the manner of Ferraz et al. (2020) chains these steps one phase at a time: calibrate a single phase (`phase_changes = integer(0)`) on its first observations, monitor the rows that follow, and start the next phase at the first index after the first run flagged by `"we_seven_same"` (column `.flag_we_seven_same` of the monitored chart, with `rules = "we_seven_same"`).

Value

A shewhart_chart object of subclass `shewhart_regression`. The `fits` slot contains a list of fitted model objects (one per phase; a phase with fewer than 3 observations reuses the previous phase's fit). The `metadata` slot additionally stores `phase_n_end` (the last within-phase position `.N` reached by each phase's fit) and `phase_sigma` (the residual sigma of each phase), which `monitor()` uses to extrapolate the last phase, together with `limits_scale` and `lower_bound`. The `sigma_hat` slot is the median of the per-phase sigmas.

The `.phase_label` column names the phases "Base", "Phase 1", ..., "Monitoring" (localised). When the index column is a `Date` (or a date-time), every phase but the last also carries its end date, e.g. "Base (until 2020-05-11)". The first index value of each phase is the date on which it starts.

References

Mandel, B. J. (1969). The Regression Control Chart. *Journal of Quality Technology*, 1(1), 1-9. doi:10.1080/00224065.1969.11980341

Wheeler, D. J., & Chambers, D. S. (1992). *Understanding Statistical Process Control* (2nd ed.). SPC Press.

Perla, R. J., Provost, S. M., Parry, G. J., Little, K., & Provost, L. P. (2020). Understanding variation in reported COVID-19 deaths with a novel Shewhart chart application. *International Journal for Quality in Health Care*, 32(10), 685-688. doi:10.1093/intqhc/mzaa069

Ferraz, C., Petenate, A. J., Leite Wanderley, A., Ospina, R., Torres, J., & Peruzzi Moreira, A. (2020). COVID-19: monitoramento por graficos de Shewhart. *Revista Brasileira de Estatistica*, 78(245), 23-41.

Box, G. E. P., & Cox, D. R. (1964). An Analysis of Transformations. *Journal of the Royal Statistical Society, Series B*, 26(2), 211-252. doi:10.1111/j.25176161.1964.tb00553.x

Examples

```
set.seed(1)
df <- data.frame(
  t = 1:60,
  y = c(1:30 * 0.5 + rnorm(30, sd = 0.5),    # phase 1: linear trend
        15 + 1:30 * 0.1 + rnorm(30, sd = 0.5)) # phase 2: shift + slowdown
```

```
)
fit <- shewhart_regression(df, value = y, index = t, model = "linear")
print(fit)
ggplot2::autoplot(fit)

# Multiplicative limits built on the log scale (Perla et al. 2020;
# Ferraz et al. 2020): base of 10 days, 7-point shift rule
br <- subset(cvd_brazil, region == "BR" & date >= as.Date("2020-03-17") &
               date <= as.Date("2020-06-20"))
fit_log <- shewhart_regression(br, value = new_deaths, index = date,
                               model = "log", limits_scale = "model",
                               phase_rule = "we_seven_same")
ggplot2::autoplot(fit_log, legend_position = "inside")
```

---

shewhart_rules_available *List available runs rules*

Description

List available runs rules

Usage

```
shewhart_rules_available()
```

Value

A tibble with columns `rule` (the key) and `description`.

Examples

```
shewhart_rules_available()
```

---

shewhart_runs *Apply runs tests to a chart object or to raw vectors*

Description

Implements the eight rules of Nelson (1984, 1985) plus a Western Electric "7 in a row" variant for backward compatibility. Returns a tidy tibble of rule violations.

Usage

```
shewhart_runs(
  x,
  rules = c("nelson_1_beyond_3s", "nelson_2_nine_same"),
  center = NULL,
  sigma = NULL
)
```

Arguments

x	Either a shewhart_chart object (most common) or a numeric vector of values. If a vector is supplied, center and sigma must also be provided.
rules	Character vector of rule keys to apply. Use shewhart_rules_available() to see all options. Default applies Nelson 1 (beyond 3 sigma) and Nelson 2 (9 same side), the most commonly recommended pair.
center, sigma	Numeric scalars or vectors. Required only when x is a numeric vector. Ignored otherwise.

Value

A tibble with columns position (integer, the index where the rule fired), rule (character key), description (character label), value (the value at that position) and severity (currently always "out_of_control"; reserved for future warning-level rules).

References

Nelson, L. S. (1984). The Shewhart Control Chart – Tests for Special Causes. *Journal of Quality Technology*, 16(4), 237-239. doi:10.1080/00224065.1984.11978921

Nelson, L. S. (1985). Interpreting Shewhart Xbar Control Charts. *Journal of Quality Technology*, 17(2), 114-117. doi:10.1080/00224065.1985.11978941

Western Electric Co. (1956). *Statistical Quality Control Handbook*.

Examples

```
set.seed(1)
x <- c(rnorm(20), 5, rnorm(20))    # one outlier at position 21
shewhart_runs(x, center = 0, sigma = 1)
```

---

shewhart_theme *Editorial-style ggplot2 theme used by every* autoplot.shewhart_*

Description

Shared across the package so charts look like one family. The visual choices are inspired by data-journalism graphics (FT, Pew Research, The Economist): off-white background, only horizontal grid lines, axis line on the data side, left-aligned title block, and tonal grey for non-data ink.

Usage

```
shewhart_theme(base_size = 10.5, base_family = "")
```

Arguments

base_size	Base font size, in points.
base_family	Base font family. Empty string uses the system default sans-serif. We do not hard-code a Google Font so the theme works in offline / CRAN-check environments.

Details

Use it from your own layers when you want a chart that matches the package's identity:

```
ggplot(d, aes(x, y)) + geom_line() + shewhart_theme()
```

Value

A `ggplot2::theme()` object.

Examples

```
library(ggplot2)
df <- data.frame(x = 1:50, y = cumsum(rnorm(50)))
ggplot(df, aes(x, y)) + geom_line() + shewhart_theme()
```

---

shewhart_u *u chart for nonconformities per unit, variable inspection size*

Description

Constructs a u chart from defect counts and a per-observation "exposure" (inspection size: square metres of fabric, hours of operation, lines of code, etc.). For constant exposure use `shewhart_c()`.

Usage

```
shewhart_u(
  data,
  defects,
  exposure,
  index = NULL,
  limits = c("3sigma", "poisson"),
  rules = c("nelson_1_beyond_3s"),
  locale = getOption("shewhart.locale", "en"),
  verbose = NULL
)
```

Arguments

data	A data frame.
defects	Tidy-eval column reference for raw defect counts.
exposure	Tidy-eval column reference for the inspection size (must be positive).
index	Optional tidy-eval column reference for the x-axis.
limits	One of "3sigma" (default; classical normal approximation) or "poisson" (exact Poisson 0.00135 / 0.99865 quantiles, recommended when c_bar < 10).
rules	Character vector of rule keys to apply. See shewhart_rules_available(). Default applies Nelson 1 and 2.
locale	One of "en", "pt", "es", "fr". Affects plot labels and informative messages.
verbose	Logical. Print progress messages? Defaults to the shewhart.verbose option.

Value

A shewhart_chart object of subclass shewhart_u.

References

Montgomery, D. C. (2019). *Introduction to Statistical Quality Control* (8th ed.). Wiley. Chapter 7.3.2.

Examples

```
set.seed(1)
df <- data.frame(
  roll     = 1:25,
  defects  = rpois(25, lambda = 4 * runif(25, 0.5, 1.5)),
  m2       = runif(25, 0.5, 1.5)    # variable inspection size
)
fit <- shewhart_u(df, defects = defects, exposure = m2, index = roll)
```

---

shewhart_xbar_r *Xbar-R control chart for rational subgroups*

Description

Constructs a paired Xbar (subgroup mean) and R (subgroup range) chart for measurements organised in rational subgroups of size 2 to 10. Sigma is estimated from the average within-subgroup range.

Usage

```
shewhart_xbar_r(
  data,
  value,
  subgroup,
  rules = c("nelson_1_beyond_3s", "nelson_2_nine_same"),
  locale = getOption("shewhart.locale", "en"),
  verbose = NULL
)
```

Arguments

data	A data frame.
value	Tidy-eval column reference for the measurement.
subgroup	Tidy-eval column reference identifying the subgroup (e.g. shift, batch, hour). All subgroups must have equal size. Subgroups are plotted and tested in order of first appearance in data (time order), not in sorted order of their labels.
rules	Character vector of rule keys to apply. See shewhart_rules_available(). Default applies Nelson 1 and 2.
locale	One of "en", "pt", "es", "fr". Affects plot labels and informative messages.
verbose	Logical. Print progress messages? Defaults to the shewhart.verbose option.

Details

Xbar-chart limits use A2(n); R-chart limits use D3(n) and D4(n). See shewhart_constants() for the tabulated values.

Value

A shewhart_chart object of subclass shewhart_xbar_r.

References

Montgomery, D. C. (2019). *Introduction to Statistical Quality Control* (8th ed.). Wiley. Chapter 6.

Shewhart, W. A. (1931). *Economic Control of Quality of Manufactured Product*. D. Van Nostrand.

Examples

```
set.seed(1)
df <- data.frame(
  batch = rep(1:25, each = 5),
  y     = rnorm(125, mean = 50, sd = 1.5)
)
fit <- shewhart_xbar_r(df, value = y, subgroup = batch)
print(fit)
```

shewhart_xbar_s *Xbar-S control chart for rational subgroups*

Description

Like shewhart_xbar_r(), but uses the subgroup standard deviation (S) instead of the range. Recommended for subgroup sizes greater than 10, or when subgroup sizes differ.

Usage

```
shewhart_xbar_s(
  data,
  value,
  subgroup,
  sigma_method = c("sbar", "pooled_sd"),
  rules = c("nelson_1_beyond_3s", "nelson_2_nine_same"),
  locale = getOption("shewhart.locale", "en"),
  verbose = NULL
)
```

Arguments

data	A data frame.
value	Tidy-eval column reference for the measurement.
subgroup	Tidy-eval column reference identifying the subgroup (e.g. shift, batch, hour). All subgroups must have equal size. Subgroups are plotted and tested in order of first appearance in data (time order), not in sorted order of their labels.
sigma_method	One of "sbar" (default; classical S-bar / c4(n)) or "pooled_sd" (pooled within-subgroup SD; preferred when subgroups have different sizes).
rules	Character vector of rule keys to apply. See shewhart_rules_available(). Default applies Nelson 1 and 2.
locale	One of "en", "pt", "es", "fr". Affects plot labels and informative messages.
verbose	Logical. Print progress messages? Defaults to the shewhart.verbose option.

Details

Xbar-chart limits use A3(n); S-chart limits use B3(n) and B4(n). When sigma_method = "pooled_sd", sigma is estimated as the pooled within-subgroup standard deviation.

Value

A shewhart_chart object of subclass shewhart_xbar_s.

References

Montgomery, D. C. (2019). *Introduction to Statistical Quality Control* (8th ed.). Wiley. Chapter 6.4.

Examples

```
set.seed(1)
df <- data.frame(
  batch = rep(1:30, each = 12),
  y     = rnorm(360, mean = 80, sd = 0.6)
)
fit <- shewhart_xbar_s(df, value = y, subgroup = batch)
print(fit)
```

SSgompertzDummy *Self-starting Gompertz with an additive dummy term*

Description

Extends the classical Gompertz form by adding a linear contribution from a dummy covariate:

$$y = \text{Asym}\, \exp(-b_2\, \exp(-b_3 x)) + \beta\, d.$$

Designed for use inside `stats::nls()`: starting values for the four parameters are computed automatically from the data, and the analytic gradient is supplied for faster, more reliable convergence.

Usage

```
SSgompertzDummy(x, dummy, Asym, b2, b3, Beta)
```

Arguments

x	Numeric vector. Independent variable.
dummy	Numeric (typically 0/1) vector of the same length as x.
Asym	Upper asymptote.
b2	Curvature parameter.
b3	Rate parameter.
Beta	Dummy effect size.

Value

A numeric vector of fitted values, with attributes for self-starting and an analytic gradient.

References

Same as `Gompertz()`.

See Also

`stats::SSgompertz()` for the standard self-starting Gompertz without a dummy term, `fit_gompertz_dummy()` for a convenience wrapper.

Examples

```
set.seed(42)
n <- 50
x <- seq(1, 10, length.out = n)
d <- rep(c(0, 1), each = n / 2)
y <- 100 * exp(-2 * exp(-0.3 * x)) + 20 * d + rnorm(n, 0, 3)
df <- data.frame(x = x, y = y, dummy = d)

fit <- nls(y ~ SSgompertzDummy(x, dummy, Asym, b2, b3, Beta), data = df)
summary(fit)
```

`summary.shewhart_chart` *Compact tibble-like summary*

Description

Compact tibble-like summary

Usage

```
## S3 method for class 'shewhart_chart'
summary(object, ...)
```

Arguments

`object`	A shewhart_chart object.
`...`	Currently unused.

Value

A list with elements `limits` and `violations`.

Examples

```
fit <- shewhart_i_mr(data.frame(y = rnorm(50)), value = y)
summary(fit)
```

tablet_weight *Pharmaceutical tablet weights*

Description

A synthetic dataset modelled on classical pharmaceutical quality control. Tablet weights are recorded in subgroups of 5 tablets each, across 25 production batches. Target weight is 250 mg with a process sigma of 1.5 mg; a small mean shift to 251.5 mg is embedded in subgroups 18-25.

Usage

```
tablet_weight
```

Format

A tibble with 125 rows and 3 columns:

subgroup Integer batch identifier (1-25).

tablet Integer tablet position within the batch (1-5).

weight Numeric tablet weight in milligrams.

Source

Synthetic. See `data-raw/build_all.R`.

See Also

`shewhart_xbar_r()`, `shewhart_xbar_s()`.

Examples

```
fit <- shewhart_xbar_r(tablet_weight, value = weight, subgroup = subgroup)

ggplot2::autoplot(fit)
```

temperature_drift *Curing oven temperature drift*

Description

A synthetic dataset of 200 sensor readings on a curing oven. The true temperature exhibits a slow linear drift superimposed on a periodic component. A classical Shewhart chart will misjudge the limits because the process is non-stationary - a regression control chart is the right tool.

Usage

```
temperature_drift
```

Format

A tibble with 200 rows and 2 columns:

minute Integer minute since start.

temp_c Numeric temperature in degrees Celsius.

Source

Synthetic. See `data-raw/build_all.R`.

See Also

`shewhart_regression()`.

Examples

```
fit <- shewhart_regression(temperature_drift,
                           value = temp_c, index = minute,
                           model = "linear")
ggplot2::autoplot(fit)
```

`tidy.shewhart_chart` *Tidy the control limits of a Shewhart chart*

Description

Returns a tibble of the chart's control limits in tall format. Each row corresponds to one line of one chart panel (CL / UCL / LCL).

Usage

```
## S3 method for class 'shewhart_chart'
tidy(x, ...)
```

Arguments

`x`	A shewhart_chart object.
`...`	Currently unused.

Value

A tibble with at least columns `chart`, `line`, `value`. For regression charts, whose limits follow a fitted curve, the tibble is in long format with one row per phase, endpoint and line: columns `chart` (`"regression"`), `.phase`, `endpoint` (`"first"` / `"last"` observation of the phase), `line` (`"CL"`, `"UCL"`, `"LCL"`) and `value`.

Examples

```
fit <- shewhart_i_mr(data.frame(y = rnorm(50)), value = y)
broom::tidy(fit)
```

Index

∗ **datasets**
bacterial_growth, 7
bottle_fill, 8
claims_p, 11
cvd_brazil, 12
cvd_recife, 14
pcb_solder, 21
tablet_weight, 54
temperature_drift, 54

as_plotly, 3
augment(), *22*
augment.shewhart_chart, 4
autoplot.shewhart_chart, 5
autoplot.shewhart_regression, 6

bacterial_growth, 7
bottle_fill, 8
box_cox, 9

calibrate, 10
calibrate(), *45*
claims_p, 11
color_hue, 12
cvd_brazil, 12
cvd_recife, *13*, 14

fit_gompertz_dummy, 15
fit_gompertz_dummy(), *52*

ggplot2::autoplot(), *3*
glance.shewhart_chart, 15
Gompertz, 16
Gompertz(), *7*, *52*

htmlwidgets::saveWidget(), *4*

iloglog, 17
iloglog(), *20*
inv_box_cox, 18
is_shewhart_chart, 19

loglog, 19
loglog(), *17*

monitor, 20
monitor(), *38*, *44*, *45*

pcb_solder, 21
plotly::ggplotly(), *4*
print.shewhart_chart, 22

rolling_sum, 22

shewhart_arl, 23
shewhart_box_cox, 24
shewhart_box_cox(), *9*, *20*
shewhart_c, 25
shewhart_c(), *21*, *48*
shewhart_capability, 26
shewhart_chart, *4, 5*, *10*, *16*, *20*, *22*, *26*, *29*, *30*, *32*, *34, 35*, *37, 38*, *40, 41*, *45*, *49–51*, *53*, *55*
shewhart_chart *(*is_shewhart_chart*)*, 19
shewhart_constants, 28
shewhart_constants(), *50*
shewhart_cusum, 28
shewhart_diagnostics, 30
shewhart_ewma, 31
shewhart_hotelling, 33
shewhart_i_mr, 34
shewhart_i_mr(), *8*
shewhart_mcusum, 36
shewhart_mewma, 37
shewhart_np, 39
shewhart_p, 40
shewhart_p(), *11*, *39*
shewhart_palette, 42
shewhart_regression, 43
shewhart_regression(), *6, 7*, *13, 14*, *55*
shewhart_rules_available, 46
shewhart_rules_available(), *23*, *25*, *35*, *40, 41*, *44*, *47*, *49–51*

shewhart_runs, 46
shewhart_theme, 47
shewhart_u, 48
shewhart_u(), *25*
shewhart_xbar_r, 49
shewhart_xbar_r(), *51*, *54*
shewhart_xbar_s, 51
shewhart_xbar_s(), *54*
SSgompertzDummy, 52
SSgompertzDummy(), *17*
stats::nls(), *15*, *52*
stats::SSgompertz(), *52*
summary.shewhart_chart, 53

tablet_weight, 54
temperature_drift, 54
tidy(), *22*
tidy.shewhart_chart, 55