Package ‘qicharts’

April 20, 2021

Version  0.5.8
Title  Quality Improvement Charts
Description  Functions for making run charts
             [Anhoej, Olesen (2014) <doi:10.1371/journal.pone.0113825>] and basic
             The main function, qic(), creates run and control charts and has a
             simple interface with a rich set of options to control data analysis
             and plotting, including options for automatic data aggregation by
             subgroups, easy analysis of before-and-after data, exclusion of one
             or more data points from analysis, and splitting charts into
             sequential time periods.
             Missing values and empty subgroups are handled gracefully.

Depends  R (>= 3.0.0)
Imports  lattice, latticeExtra, graphics, grDevices, stats, scales,
         ggplot2 (>= 2.0.0),
Suggests  knitr, rmarkdown
VignetteBuilder  knitr
License  GPL-3
Encoding  UTF-8
RoxygenNote  7.1.1
NeedsCompilation  no
Author  Jacob Anhoej [aut, cre],
        Timo Roeder [ctb]
Maintainer  Jacob Anhoej <jacob@anhoej.net>
Repository  CRAN
Date/Publication  2021-04-20 12:20:03 UTC

R topics documented:

paretochart ............................................  2
Description

Creates a pareto chart from a categorical variable

Usage

paretochart(
  x,
  main,
  ylab = "Frequency",
  xlab = "",
  cumperc.by = 20,
  cex = 0.8,
  ...
)

Arguments

  x  Categorical vector to be plotted
  main Plot title
  ylab Label on y axis
  xlab Label on x axis
  cumperc.by Grid interval
  cex  Number indicating the amount by which text and symbols should be magnified.
  ... Further arguments to plot function

Value

A table of frequencies and percentages from the pareto analysis

Author(s)

Jacob Anhoej

Examples

x <- rep(LETTERS[1:9], c(256, 128, 64, 32, 16, 8, 4, 2, 1))
paretochart(x)
### plot.qic

**Plot qic object**

#### Description

Plot qic object

#### Usage

```r
## S3 method for class 'qic'
plot(x, y = NULL, ...)
```

#### Arguments

- `x` List object returned from the qic() function.
- `y` Ignored. Included for compatibility with generic plot function.
- `...` Further arguments to plot function.

#### Value

Creates a qic plot.

#### Examples

```r
y <- rnorm(24)
p <- qic(y, plot.chart = FALSE)
plot(p)
```

### qic

**Quality improvement charts**

#### Description

Run and control charts for quality improvement and control

#### Usage

```r
qic(
  y,
  n,
  x,
  data,
  chart = c("run", "i", "mr", "xbar", "s", "t", "p", "c", "u", "g"),
  notes = NULL,
  ...)
```
Arguments

y Numeric vector of counts or measures to plot. Mandatory.
n Numeric vector of sample sizes. Mandatory for P and U charts.
x Subgrouping vector used for aggregating data and making x-labels. Mandatory for Xbar and S charts.
data Data frame containing variables.
chart Type of control chart. Possible types are:
  • "run": run chart (default).
  • "i": individuals chart.
  • "mr": moving range chart.
  • "xbar": sample average chart.
  • "s": sample standard deviation chart.
  • "t": time between events chart.
  • "p": proportions chart.
- "c": counts chart.
  - "u": rates chart.
  - "g": cases between events chart.

notes
Character vector of notes to be added to individual data points.

c1
Value specifying the center line (if known). Must be of length one or same as number of subgroups (for variable center line).

agg.fun
String specifying the aggregate function if there is more than one value per subgroup. Possible values are ‘mean’ and ‘sum’. Only relevant if you want to aggregate count data with run charts or I charts. If agg.fun = ‘sum’, the n argument (if provided) will be ignored.

ylim
Range of y axis limits.

target
Value specifying a target line to plot.

direction
Value indication direction of improvement, 0 (down) or 1 (up).

freeze
Number identifying the last data point to include in calculations of center and limits (ignored if breaks argument is given).

breaks
Numeric vector of break points. Useful for splitting graph in two or more sections with separate center line and control limits.

exclude
Numeric vector of data points to exclude from calculations of center and control lines.

negy
Logical value, if TRUE, the y axis is allowed to be negative (only relevant for I and Xbar charts).

dots.only
Logical value. If TRUE, data points are not connected by lines and runs analysis is not performed. Useful for comparison and funnel plots.

multiply
Integer indicating a number to multiply y axis by, e.g. 100 for percents rather than proportions.

prime
Logical value, if TRUE, control limits incorporate between-subgroup variation as proposed by Laney (2002). This is recommended for data involving very large sample sizes. Only relevant for P and U charts.

standardised
Logical value, if TRUE, creates a standardised control chart, where points are plotted in standard deviation units along with a center line at zero and control limits at 3 and -3. Only relevant for P, U and Xbar charts.

x.format
Date format of x axis labels. See ?strftime for date formats.

nint
Number indicating (approximately) the desired number of tick marks on the x axis.

cex
Number indicating the amount by which text and symbols should be magnified.

main
Character string specifying the title of the plot.

xlab
Character string specifying the x axis label.

ylab
Character string specifying the y axis label.

sub
Character string specifying a subtitle to be printed in the lower left corner of the plot.

decimals
Integer indicating the number of decimals shown for center and limits on the plot. Default behaviour is smart rounding to at least two significant digits.
pre.text Character string labelling pre-freeze period
post.text Character string labelling post-freeze period
llabs Character vector with four elements specifying labels for lower control limit, centre line, upper control limit and target line respectively
runvals Logical value, if TRUE, prints statistics from runs analysis on plot.
linevals Logical value, if TRUE, prints values for center and control lines on plot.
plot.chart Logical value, if TRUE, prints plot.
print.out Logical value, if TRUE, prints return value
... Further arguments to plot function.

Details

If chart is not specified, qic() plots a run chart. Non-random variation will be marked by a dashed, yellow center line (the median) if either the longest run of data points above or below the median is longer than predicted or if the graph crosses the median fewer times than predicted (see references for details).

Only the y argument giving the count or measure of interest is mandatory for a run chart. If a denominator argument, n, is given, y/n will be plotted. If a subgrouping argument, x, is given, \( \text{sum}(y)/\text{sum}(n) \), within each subgroup will be plotted. This behaviour can be modified using the agg.fun argument.

With controlcharts, data aggregation by subgroups is handled according to chart type. For P, U, and I charts, data are aggregated as described for the run chart. For the C chart, the sum of counts, \( \text{sum}(y) \), within each subgroups will be plotted.

For Xbar and S charts, the subgrouping argument, x, is mandatory. However, the sample size argument, n, is irrelevant and will be ignored.

The subgrouping argument, x, is irrelevant for T and G charts, and, if given, an error will occur if any subgroup has more than one element.

If more than one note is present within any subgroup, the first note (alphabetically) is chosen.

If both prime and standardised are TRUE, points are plotted in units corresponding to Laney’s modified "standard deviation", which incorporates the variation between subgroups.

Value

A list of of class qic containing values and parameters of the qic plot.

References

Runs analysis:


Calculation of control limits:


Examples

```r
set.seed(1)
# Run chart of 24 samples of a random continuous variable
# with an approximate mean = 12 and standard deviation = 3.
y <- rnorm(24, 12, 3)
qic(y)

# Add subgroup vector (dates) and a target
x <- seq.Date(as.Date("2013-08-04"), by = "week", length = 24)
qic(y, x = x, target = 16)

# Individuals control chart
qic(y, x = x, chart = "i")

# Xbar control chart, sample size = 5
y <- rnorm(5 * 24)
x <- rep(x, 5)
qic(y, x = x, chart = "xbar")

# Create data frame with counts and sample sizes by week
d <- data.frame(week = seq.Date(as.Date("2013-08-04"),
                                by = "week",
                                length = 36),
y = c(rbinom(24, 20, 0.5), rbinom(12, 20, 0.8)),
n = round(rnorm(36, 20, 2)))

# Proportions control chart
qic(y, n, x = week, data = d[1:24,], chart = "p")

# Introduce change in process performance
qic(y, n, x = week, data = d, chart = "p")

# Freeze baseline to first 24 samples
qic(y, n, x = week, data = d, chart = "p", freeze = 24)
```
# Break control chart before and after change
qic(y, n, x = week, data = d, chart = 'p', breaks = 24)

# Introduce extreme sample value and notes
d$a <- ''
d$a[30] <- 'Extreme value'
d$y[30] <- 1
qic(y, n, x = week, data = d, chart = 'p',
    breaks = 24,
    notes = a)

# Exclude value from calculations
d$a[30] <- 'Value excluded from calculations'
qic(y, n, x = week, data = d, chart = 'p',
    breaks = 24,
    notes = a,
    exclude = 30)

---

summary.tcc

## Summarise Trellis Control Charts

### Description
Summary function for tcc objects.

### Usage
```r
## S3 method for class 'tcc'
summary(object, ...)
```

### Arguments
- **object**: tcc object
- **...**: Ignored. Included for compatibility with generic summary function.

### Value
A data frame with summary statistics of the tcc object.

### Examples
```r
# Build data frame for example
d <- data.frame(x = rep(1:24, 4),
    mo = (rep(seq(as.Date('2014-1-1'),
        length.out = 24,
        by = 'month'),
        4)),
    n = rbinom(4 * 24, 100, 0.5),
```
d = round(runif(4 * 24, 90, 110)),
g1 = rep(c('a', 'b'), each = 48),
g2 = rep(c('A', 'B'), each = 24))

# P chart
p <- tcc(n, d, mo, g1 = g1, g2 = g2, breaks = 12, data = d, chart = 'p')
plot(p)
summary(p)

---

tcc

Trellis Control Charts

Description

Run and control charts for multivariate data in trellis (grid) layout.

Usage

tcc(
n,
d,
x,
g1,
g2,
breaks,
notes,
data,
chart = c("run", "i", "mr", "xbar", "s", "t", "p", "c", "u", "g"),
multiply = 1,
freeze = NULL,
exclude,
target = NA,
n.sum = FALSE,
y.neg = TRUE,
y.percent = FALSE,
y.expand = NULL,
x.pad = 1,
x.date.format = NULL,
c1.lab = TRUE,
c1.decimals = NULL,
main,
xlab = "Subgroup",
ylab = "Value",
subtitle = NULL,
caption = NULL,
cex = 1,
pex = 1,
prime = TRUE,
flip = FALSE,
dots.only = FALSE,
print.summary = FALSE,
...
)

Arguments

n       Numerator, numeric vector of counts or measures to plot. Mandatory.
d       Denominator, numeric vector of subgroup sizes. Mandatory for P and U charts.
x       Subgrouping vector used for aggregating data by subgroup and making x-labels. Mandatory for Xbar and S charts.
g1      Grouping vector 1 used for trellis layout (facets).
g2      Grouping vector 2 used for trellis layout (facets).
breaks  Numeric vector of break points. Useful for splitting graph in two or more sections with separate center line and control limits.
notes   Character vector of notes to be added to individual data points.
data    Data frame containing variables.
chart   Type of control chart. Possible types are:
        • "run": run chart (default).
        • "i": individuals chart.
        • "mr": moving range chart.
        • "xbar": sample average chart.
        • "s": sample standard deviation chart.
        • "t": time between events chart.
        • "p": proportions chart.
        • "c": counts chart.
        • "u": rates chart.
        • "g": cases between events chart.
multiply Integer indicating a number to multiply y axis by, e.g. 100 for percents rather than proportions. See also y.percent argument.
freeze  Number identifying the last data point to include in calculations of center and limits (ignored if breaks argument is given).
exclude Numeric vector of data points to exclude from runs analysis and calculations of center and control lines (same for each facet).
target  Numeric value indicating a target value to be plotted as a horizontal line (same for each facet).
n.sum   Logical value indicating whether the mean (default) or sum of numerator (n argument) per subgroup should be plotted. Only relevant for run, C, and I charts with multiple counts per subgroup.
y.neg   Logical value. If TRUE (default), the y axis is allowed to be negative (only relevant for I and Xbar charts).
y.percent Logical. If TRUE, formats y axis labels as percent.

y.expand Numeric value to include in y axis. Useful e.g. for beginning y axis at zero.

x.pad Number indicating expansion of x axis to make room for center line labels.

x.date.format Date format of x axis labels. See strftime() for possible date formats.

c.lab Logical value. If TRUE (default), plots center line labels.

cl.decimals Number of decimals on center line labels.

main Character string specifying the title of the plot.

xlab Character string specifying the x axis label.

ylab Character string specifying the y axis label.

subtitle Character string specifying the subtitle.

caption Character string specifying the caption.

cex Number indicating the amount by which text should be magnified.

pex Number indicating the amount by which plotting symbols should be magnified.

prime Logical value, If TRUE (default unless dots.only = TRUE), control limits incorporate between-subgroup variation as proposed by Laney (2002). Only relevant for P and U charts.

flip Logical. If TRUE rotates the plot 90 degrees.

dots.only Logical value. If TRUE, data points are not connected by lines, prime is forced to be FALSE, and runs analysis is not performed. Useful for comparison and funnel plots.

print.summary Logical. If TRUE, prints summary of tcc object.

... Further arguments to ggplot function.

Details
tcc() is a wrapper function for ggplot2() that makes multivariate run and control charts. It takes up to two grouping variables for multidimensional trellis plots.

Note that, in contrast to the qic() function, the prime argument defaults to TRUE, which means that control limits of P and U charts by default incorporate between-subgroup variation as proposed by Laney (2002).

Value

An object of class ggplot.

References

Runs analysis:


Calculation of control limits:


Examples

```r
# Run chart of 24 random normal variables
tcc(rnorm(24))

# Build data frame for examples
d <- data.frame(x = rep(1:24, 4),
    mo = (rep(seq(as.Date('2014-1-1'), length.out = 24, by = 'month'), 4)),
    n = rbinom(4 * 24, 100, 0.5),
    d = round(runif(4 * 24, 90, 110)),
    g1 = rep(c('a', 'b'), each = 48),
    g2 = rep(c('A', 'B'), each = 24))

# Run chart with two grouping variables
tcc(n, d, mo, g1 = g1, g2 = g2, data = d)

# P chart
tcc(n, d, mo, g1 = g1, g2 = g2, data = d, chart = 'p')

# P chart with baseline fixed to the first 12 data points
tcc(n, d, mo, g1 = g1, g2 = g2, data = d, chart = 'p', freeze = 12)

# P chart with two breaks and summary output
tcc(n, d, mo, g1 = g1, g2 = g2, data = d, chart = 'p',
    breaks = c(12, 18), print.summary = TRUE)
```
Description

Run charts for multivariate data in trellis (grid) layout.

Usage

```
trc(
x, chart = c("run", "i"),
xscale = "same",
yscale = "same",
dec = NULL,
 xpad = 0.1,
pch = 20,
cex = 0.7,
gap = 0.5,
target = NA,
...
)
```

Arguments

- **x**: Formula object to plot. The formula is of the form `y ~ x | g1 + g2 + ...`, indicating that plots of `y` (on the y-axis) versus `x` (on the x-axis) should be produced conditional on the variables `g1`, `g2`.
- **chart**: Type of chart: 'run' or 'i'.
- **xscale**: Scaling of x-axes: 'same' or 'free'.
- **yscale**: Scaling of y-axes: 'same' or 'free'.
- **dec**: Number of decimals of median value. The default behaviour (smart rounding to at least two significant digits) should be satisfactory in most cases.
- **xpad**: Number specifying the fraction by which to extend the x-axis in order to make space for the median label.
- **pch**: Plotting character.
- **cex**: Number indicating the magnification of plotting character.
- **gap**: Number indicating spacing between panels.
- **target**: Value specifying a target line to plot.
- ...: Further arguments to `xyplot`.

Details

This function is a wrapper for `xyplot` from the `lattice` package. Some useful arguments from `xyplot` are `main`, `ylab`, `xlab`, and `layout`. 
Value

Returns an object of class "trellis".

See Also

xyplot, qic

Examples

# Trellis run chart on 1 conditioning variable
d1 <- data.frame(y = rnorm(96, 12, 3),
                 expand.grid(x = 1:24,
                              g = LETTERS[1:4]))
trc(y ~ x | g, data = d1, main = 'Trellis run chart')

# Add target line
trc(y ~ x | g, data = d1, main = 'Trellis run chart', target = 20)

# Trellis run chart on 2 conditioning variables
d2 <- data.frame(y = rnorm(144, 12, 3),
                 expand.grid(x = seq.Date(as.Date('2014-1-1'),
                                 by = 'week',
                                 length.out = 24),
                              g1 = LETTERS[1:3],
                              g2 = letters[1:2]))
trc(y ~ x | g1 + g2, data = d2, main = 'Trellis run chart')

# Introduce a shift in process performance
trc(y ~ x | g1 + g2, data = d2, main = 'Trellis run chart')

# Make I chart
trc(y ~ x | g1 + g2, data = d2, main = 'Trellis run chart', chart = 'i')
Index

lattice, 13
paretochart, 2
plot.qic, 3
qic, 3, 14
summary.tcc, 8
tcc, 9
trc, 13
xyplot, 13, 14