Package ‘rgr’

February 20, 2015

Type Package
Title Applied Geochemistry EDA
Version 1.1.10
Date 2015-01-02
Author Robert G. Garrett
Maintainer Robert G. Garrett <garrett@NRCan.gc.ca>
Depends MASS, fastICA
Suggests akima
Description Geological Survey of Canada (GSC) functions for exploratory data analysis with applied geochemical data, with special application to the estimation of background ranges and identification of outliers, 'anomalies', to support mineral exploration and environmental studies. Additional functions are provided to support analytical data QA/QC, ANOVA for investigations of field sampling and analytical variability, and utility tasks. NOTE: function caplot for concentration-area plots employs package akima, however, akima is only licensed for not-for-profit use. Therefore, not-for-profit users of rgr will have to independently make package akima available through library(akima); and use of function caplot by for-profit users will fail.
License GPL-2
NeedsCompilation no
Repository CRAN
Date/Publication 2015-01-04 00:29:23

R topics documented:

  rgr_1.1.10-package .............................................. 4
  ad.plot1 .......................................................... 5
  ad.plot2 .......................................................... 7
  ad.plot3 .......................................................... 8
  ad.plot4 .......................................................... 10
  ad.test ........................................................... 12
  alr ................................................................. 13
  alts2dups ......................................................... 15
  anova1 ............................................................ 16
R topics documented:

anova2 ................................................................. 18
bwplots ................................................................. 19
bwplots.by.var ....................................................... 23
bxplot ................................................................. 26
caplot ................................................................. 29
cat2list ............................................................... 32
clr ................................................................. 33
cnpplt ............................................................... 35
crm.plot ............................................................. 37
crm.plot.new ......................................................... 39
crm.test ............................................................. 41
crm.test.new ........................................................ 42
cutter ................................................................. 42
df.test ................................................................. 43
display.ascii.o ....................................................... 44
display.lty ........................................................... 44
display.marks ......................................................... 45
display.rainbow ....................................................... 45
expit ................................................................. 46
fences ................................................................. 47
fences.summary ....................................................... 49
fix.test ............................................................... 51
fix.test.asis ........................................................ 51
framework.stats ..................................................... 52
framework.summary ................................................. 53
gx.2dproj ............................................................ 55
gx.2dproj.plot ....................................................... 58
gx.add.chisq ........................................................ 60
gx.adjr2 ............................................................. 61
gx.cnnpplts .......................................................... 62
gx.cnnpplts.setup .................................................. 64
gx.ecdf .............................................................. 65
gx.fractile ........................................................... 67
gx.hist ............................................................... 69
gx.hypergeom ......................................................... 71
gx.ks.test ........................................................... 72
gx.lm.vif ............................................................. 74
gx.md.display ....................................................... 76
gx.md.gait ........................................................... 77
gx.md.gait.closed ................................................... 80
gx.md.plot .......................................................... 83
gx.md.plt0 ........................................................... 85
gx.md.print .......................................................... 86
gx.mva .............................................................. 88
gx.mva.closed ....................................................... 91
gx.mvalloc .......................................................... 94
gx.mvalloc.closed .................................................. 96
gx.mvalloc.print ..................................................... 99
<table>
<thead>
<tr>
<th>R topics documented</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>gx.pairs4parts</td>
<td>100</td>
</tr>
<tr>
<td>gx.pearson</td>
<td>102</td>
</tr>
<tr>
<td>gx.plot2parts</td>
<td>103</td>
</tr>
<tr>
<td>gx.quantile</td>
<td>105</td>
</tr>
<tr>
<td>gx.rma</td>
<td>106</td>
</tr>
<tr>
<td>gx.robmva</td>
<td>108</td>
</tr>
<tr>
<td>gx.robmva.closed</td>
<td>112</td>
</tr>
<tr>
<td>gx.rotate</td>
<td>115</td>
</tr>
<tr>
<td>gx.rqpca.loadplot</td>
<td>117</td>
</tr>
<tr>
<td>gx.rqpca.plot</td>
<td>118</td>
</tr>
<tr>
<td>gx.rqpca.print</td>
<td>121</td>
</tr>
<tr>
<td>gx.rqpca.screeplot</td>
<td>123</td>
</tr>
<tr>
<td>gx.runs</td>
<td>124</td>
</tr>
<tr>
<td>gx.scores</td>
<td>125</td>
</tr>
<tr>
<td>gx.sm</td>
<td>126</td>
</tr>
<tr>
<td>gx.sort</td>
<td>128</td>
</tr>
<tr>
<td>gx.sort.df</td>
<td>129</td>
</tr>
<tr>
<td>gx.spearman</td>
<td>130</td>
</tr>
<tr>
<td>gx.stats</td>
<td>131</td>
</tr>
<tr>
<td>gx.subset</td>
<td>133</td>
</tr>
<tr>
<td>gx.summary</td>
<td>135</td>
</tr>
<tr>
<td>gx.summary.groups</td>
<td>136</td>
</tr>
<tr>
<td>gx.summary.mat</td>
<td>138</td>
</tr>
<tr>
<td>gx.summary1</td>
<td>139</td>
</tr>
<tr>
<td>gx.summary2</td>
<td>141</td>
</tr>
<tr>
<td>gx.triples.aov</td>
<td>142</td>
</tr>
<tr>
<td>gx.triples.fgx</td>
<td>143</td>
</tr>
<tr>
<td>gx.vm</td>
<td>145</td>
</tr>
<tr>
<td>ilr</td>
<td>146</td>
</tr>
<tr>
<td>ilr.stab</td>
<td>148</td>
</tr>
<tr>
<td>inset</td>
<td>149</td>
</tr>
<tr>
<td>inset.exporter</td>
<td>151</td>
</tr>
<tr>
<td>kola.c</td>
<td>153</td>
</tr>
<tr>
<td>kola.o</td>
<td>154</td>
</tr>
<tr>
<td>logit</td>
<td>155</td>
</tr>
<tr>
<td>ltdl.fix</td>
<td>157</td>
</tr>
<tr>
<td>ltdl.fix.df</td>
<td>159</td>
</tr>
<tr>
<td>map.eda7</td>
<td>161</td>
</tr>
<tr>
<td>map.eda8</td>
<td>164</td>
</tr>
<tr>
<td>map.tags</td>
<td>166</td>
</tr>
<tr>
<td>map.z</td>
<td>168</td>
</tr>
<tr>
<td>ms.data1</td>
<td>172</td>
</tr>
<tr>
<td>ms.data2</td>
<td>172</td>
</tr>
<tr>
<td>ms.data3</td>
<td>173</td>
</tr>
<tr>
<td>orgrady</td>
<td>174</td>
</tr>
<tr>
<td>orgrady.mat2open</td>
<td>175</td>
</tr>
<tr>
<td>orthonorm</td>
<td>176</td>
</tr>
<tr>
<td>remove.na</td>
<td>178</td>
</tr>
</tbody>
</table>
The GSC (Geological Survey of Canada) Applied Geochemistry EDA Package

**Description**

Functions for Exploratory Data Analysis with applied geochemical data.

**Details**

This package supports the display and analysis of applied geochemical survey data, particularly in the context of estimating the ranges of background variation due to natural phenomena and the identification of outliers that may be due to natural processes or anthropogenic contamination. Functions are provided for use with univariate and multivariate data, in the latter context tools are provided for compositional, constant sum, data. Additional functions support analytical data QA/QC, ANOVA for investigations of field sampling and analytical variability, and utility tasks. NOTE: function caplot for concentration-area plots employs package akima, however, akima is only licensed for not-for-profit use. Therefore, not-for-profit users of rgr will have to independently make package akima available through library(akima); and use of function caplot by for-profit users will fail.

| Package:   | rgr_1.1.10 |
| Type:      | Package    |
| Version:   | 1.0        |
| Date:      | 2015-01-02 |
| License:   | GPL-2      |
Ad.plot1

Author(s)
Robert G. Garrett <garrett@NRCan.gc.ca>

References

Description
Function to plot the results of analytical duplicate analyses as the percent absolute difference between duplicates relative to their means, or as the Relative Standard Deviation of the individual duplicate. Classical and robust estimates of the arithmetic Relative Standard Deviation (%) and the mean/median to which they apply are displayed based on the pooled estimate of analytical variability from the duplicates. If the duplicate data span more than 1.5 orders of magnitude this estimate is unreliable due to heterogeneity of variance considerations (heteroscedasticity). The x-axis data may either present the duplicates in the order in which they occur in the data, usually a time-series, as the duplicate means, or as the ratio of the duplicate mean to the lower detection limit if it is provided. Optionally the x-axis may be scaled logarithmically if the range of the data requires. If there is a target acceptance level it may be provided and will be displayed as a red dashed line on the plot. For data stored in alternate forms from that expected by this function use ad.plot2.

Usage
ad.plot1(x1, x2, xname = deparse(substitute(x1)), if.order = TRUE, if.rsds = FALSE, ldl = NULL, ad_tol = NULL, log = FALSE, ...)

Arguments
x1 a column vector from a matrix or data frame, x1[1], ..., x1[n].
x2 another column vector from the matrix or data frame, x2[1], ..., x2[n]. x1 and x2 must be of identical length, n, where x2 is a duplicate measurement of x1.
xname

A title can be displayed with the plot and results, e.g., \texttt{xname = "Cu (mg/kg)"}. If this field is undefined the character string for \texttt{x} is used as a default.

\textbf{if.order}

By default the analytical duplicate results are plotted in the order in which they occur in the data file, this usually corresponds to date of analysis. Alternately, setting \texttt{if.order = FALSE} results in the results being plotted against their means.

\textbf{if.rsd}

By default the absolute difference between the duplicates expressed as a percentage of their mean is plotted on the y-axis. If it is required to plot the individual duplicate relative standard deviations (RSDs), set \texttt{if.rsd = TRUE}.

ldl

By default the x-axis is defined by the measurement units. If it is desired to express the duplicate means as a ratio to the lower detection limit (ldl) of the analytical procedure, then set \texttt{ldl = \textquote{ldl}} in measurement units.

ad.tol

Optionally a tolerance level may be provided for the maximum acceptable percent absolute relative difference between duplicates, in which case a red dotted line is added to the plot. When \texttt{if.rsd = TRUE} the value of \texttt{ad.tol} is interpreted as the maximum acceptable individual duplicate relative standard deviation.

log

Optionally the x-axis of the plot employing duplicate means may be plotted with logarithmic scaling, if so, set \texttt{log = TRUE}.

... any additional arguments to be passed to the \texttt{plot} function for titling, etc.

\textbf{Details}

If the data are as a single concatenated vector from a matrix or data frame as \texttt{x1[1], \ldots, x1[n]} followed by \texttt{x[n+1], \ldots, x[2n]}, or alternated as \texttt{x[1]} and \texttt{x[2]} being a pair through to \texttt{x[2*i+1]} and \texttt{x[2*i+2]} for the \texttt{i} in \texttt{1:n} duplicate pairs use function \texttt{ad.plot2}.

For examples see \texttt{ad.plot2} as Geological Survey of Canada National Geochemical Reconnaissance survey data are not stored in this format. This function is present as the graphical equivalent to \texttt{anova1}.

\textbf{Note}

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see \texttt{ltdl.fix.df}.

\textbf{Author(s)}

Robert G. Garrett

\textbf{See Also}

\texttt{ad.plot2, ad.plot3, ltdl.fix.df}
Plot Results of Analytical Duplicate Analyses, Alternate Input

Description

Function to prepare data stored in alternate forms from that expected by function `ad.plot1` for its use, for further details see `x` in Arguments below. The data will be plotted as the percent absolute difference between duplicates relative to their accession order or their means.

Usage

```r
ad.plot2(x, xname = deparse(substitute(x)), if.order = TRUE,
         if.rsds = FALSE, ldl = NULL, ad.tol = NULL, log = FALSE,
         ifalt = FALSE, ...)```

Arguments

- `x` a column vector from a matrix or data frame, `x[1], ..., x[2*n]`. The default is that the first `n` members of the vector are the first measurements and the second `n` members are the duplicate measurements. If the measurements alternate, i.e., duplicate pair 1 measurement 1 followed by measurement 2, etc., set `ifalt = TRUE`.

- `xname` a title can be displayed with the plot and results, e.g., `xname = "Cu (mg/kg)"`. If this field is undefined the character string for `x` is used as a default.

- `if.order` by default the analytical duplicate results are plotted in the order in which they occur in the data file, this usually corresponds to date of analysis in a time-series. Alternately, setting `if.order = FALSE` results in the individual duplicate results being plotted against their means.

- `if.rsds` by default the absolute difference between the duplicates expressed as a percentage of their mean is plotted on the y-axis. If it is required to plot the relative standard deviations (RSDs), set `if.rsds = TRUE`.

- `ldl` by default the x-axis is defined by the measurement units. If it is desired to express the duplicate means as a ratio to the lower detection limit (ldl) of the analytical procedure, then set `ldl = '1dl'` in measurement units.

- `ad.tol` optionally a tolerance level may be provided for the maximum acceptable percent absolute relative difference between duplicates, in which case a red dotted line is added to the plot.

- `log` optionally the x-axis of the plot employing duplicate means may be plotted with logarithmic scaling, if so, set `log = TRUE`.

- `ifalt` set `ifalt = TRUE` to accommodate alternating sets of paired observations.

- `...` any additional arguments to be passed to the `plot` function for titling, etc.
Details

Data may be as a single concatenated vector from a matrix or data frame as x1[1], ..., x1[n] followed by x[n+1], ..., x[2n], or alternated as x[1] and x[2] being a pair through to x[2*i+1] and x[2*i+2] for the i in 1:n duplicate pairs, see ifalt.

If the data are as n duplicate pairs, x1 and x2, use function ad.plot1.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df.

Author(s)

Robert G. Garrett

See Also

ad.plot1, ad.plot4, ltdl.fix.df

Examples

```r
## Make test data available
data(ad.test)attach(ad.test)

## Plot analytical duplicate analyses as a time-series
ad.plot2(Cu, ifalt = TRUE)

## Plot analytical duplicate analyses versus duplicate means,
## annotating more appropriately, with a 20% maximum tolerance
ad.plot2(Cu, "Cu (mg/kg)", if.order = FALSE, ad.tol = 20, ifalt = TRUE)

## Detach test data
detach(ad.test)
```

ad.plot3  
*Plot Results of Analytical Duplicate Analyses as Ratios*

Description

Function to plot the results of analytical duplicate analyses as ratios vs. their means with logarithmic scaling. Classical and robust estimates for the mean/median and Relative Standard Deviations (%) are displayed. In addition to displaying this information the 95% standard error interval on the mean ratio are estimated and used to test that the mean ratio is not significantly different from one. A situation that would indicate severe systematic drift between the analytical duplicates, something that could happen if the duplicates were analysed in a separate batch at a later date, or were analysed as a group at the beginning or end of an analytical batch. The 95% confidence bounds for the duplicate ratios are plotted as dotted lines on the plot. If the lower detection for the analyses is
defined, and it falls within the range of the duplicate means, it will be plotted as a vertical dotted line. To facilitate comparison between measurements, or measurements by different methods, the range of the ratio display can be defined. Optionally, the locator may be deployed so that the ratios for estimating the 95% confidence bounds on any value may be placed on the plot in a location of the user’s choice. Optionally the ratio plot may be supplemented by an CPP of the ratios, in which case they are presented as side-by-side plots. For data stored in alternate forms from that expected by this function use \texttt{ad.plot4}.

\textbf{Usage}

\begin{verbatim}
ad.plot3(x1, x2, xname = deparse(substitute(x1)), if.order = TRUE, ad.tol = NULL,
ldl = NULL, maxrat = NULL, if.text = FALSE, if.cpp = FALSE, ...)
\end{verbatim}

\textbf{Arguments}

- \texttt{x1}  
  a column vector from a matrix or data frame, \(x1[1], \ldots, x1[n]\).

- \texttt{x2}  
  another column vector, \(x2[1], \ldots, x2[n]\). \(x1\) and \(x2\) must be of identical length, where \(x2\) is a duplicate measurement of \(x1\).

- \texttt{xname}  
  a title can be displayed with the plot and results, e.g., \texttt{xname = "Cu (mg/kg)"}. If this field is undefined the character string for \(x\) is used as a default.

- \texttt{if.order}  
  by default the analytical duplicate results are plotted in the order in which they occur in the data file, this usually corresponds to date of analysis. Alternately, setting \texttt{if.order = FALSE} results in the results being plotted against their means.

- \texttt{ad.tol}  
  a tolerance level may be provided for the maximum acceptable percent ratio between duplicates when they are plotted sequentially, i.e. \texttt{if.order = TRUE}, in which case red dotted lines are added to the plot.

- \texttt{ldl}  
  if the lower detection limit (ldl) of the analytical procedure is provided and falls within the range of the duplicate means it will be plotted as a vertical black dotted line. If another colour is required, e.g., red, set \texttt{col = 2}. See function \texttt{display.ly} for the range of available colours.

- \texttt{maxrat}  
  optionally the maximum ratio for the plot axis, the minimum is computed as \(1/\text{maxrat}\), may be set. This is useful when making comparisons between elements or between measurement methods and ensures similar scaling. Otherwise \texttt{maxrat} is determined from the data.

- \texttt{if.text}  
  if set \texttt{if.text = TRUE} the locator is deployed so that the ratios for estimating the 95% confidence bounds on any value may be placed on the ratio plot.

- \texttt{if.cpp}  
  if set \texttt{if.cpp = TRUE} the ratio plot is displayed side-by-side with an Normal Cumulative Probability Plot (CPP) of the ratios. In this instance \texttt{if.text} is not an option.

- \texttt{...}  
  any additional arguments to be passed to the \texttt{plot} function for titling, etc.

\textbf{Details}

If the data are as a single concatenated vector from a matrix or data frame as \(x1[1], \ldots, x1[n]\) followed by \(x[n+1], \ldots, x[2n]\), or alternated as \(x[1]\) and \(x[2]\) being a pair through to \(x[2*i+1]\) and \(x[2*i+2]\) for the \(i\) in 1:n duplicate pairs use function \texttt{ad.plot4}.
For examples see `ad.plot4` as Geological Survey of Canada National Geochemical Reconnaissance survey data are not stored in this format. This function is present as the graphical equivalent to `anova1`.

**Note**

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

**Author(s)**

Robert G. Garrett

**See Also**

`ad.plot4, ad.plot2, ltdl.fix.df`

---

**ad.plot4**  
*Plot Results of Analytical Duplicate Analyses as Ratios, Alternate Input*

**Description**

Function to prepare data stored in alternate forms from that expected by function `ad.plot3` for its use, for further details see `x` in Arguments below. The data will be plotted as ratios, i.e. `dup1/dup2`, with logarithmic scaling relative to their means.

**Usage**

```
ad.plot4(x, xname = deparse(substitute(x)), if.order = TRUE, ad.tol = NULL, ifalt = FALSE, ldl = NULL, maxrat = NULL, if.text = FALSE, if.cpp = FALSE, ...)
```

**Arguments**

- **x**
  
a column vector from a matrix or data frame, `x[1]`, ..., `x[2*n]`. The default is that the first `n` members of the vector are the first measurements and the second `n` members are the duplicate measurements. If the measurements alternate, i.e. duplicate pair 1 measurement 1 followed by measurement 2, etc., set `ifalt = TRUE`.

- **xname**
  
a title can be displayed with the plot and results, e.g., `xname = "Cu (mg/kg)"`. If this field is undefined the character string for `x` is used as a default.

- **if.order**
  
by default the analytical duplicate results are plotted in the order in which they occur in the data file, this usually corresponds to date of analysis. Alternately, setting `if.order = FALSE` results in the results being plotted against their means.
Optionally a tolerance level may be provided for the maximum acceptable percent ratio between duplicates, in which case a red dotted line is added to the plot.

If an alternating sets of paired observations are accommodated, set `ifalt = TRUE`.

If the lower detection limit (ldl) of the analytical procedure is provided and falls within the range of the duplicate means it will be plotted as a vertical black dotted line. If another colour is required, e.g., red, set `col = 2`. See function `display.1ty` for the range of available colours.

Optionally the maximum ratio for the plot axis, the minimum is computed as `1/maxrat`, may be set. This is useful when making comparisons between elements or between measurement methods and ensures similar scaling. Otherwise `maxrat` is determined from the data.

If set `if.text = TRUE` the locator is deployed so that the ratios for estimating the 95% confidence bounds on any value may be placed on the ratio plot.

If set `if.cpp = TRUE` the ratio plot is displayed side-by-side with an Normal Cumulative Probability Plot (CPP) of the ratios. In this instance `if.text` is not an option.

Any additional arguments to be passed to the `plot` function for titling, etc.

Details

Data may be as a single concatenated vector from a matrix or data frame as `x[1]`,..., `x[n]` followed by `x[n+1]`,..., `x[2*n]`, or alternated as `x[1]` and `x[2]` being a pair through to `x[2*i+1]` and `x[2*i+2]` for the `i` in 1:n duplicate pairs, see `ifalt`.

If the data are as n duplicate pairs, `x1` and `x2`, use function `ad.plot3`.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Author(s)

Robert G. Garrett

See Also

`ad.plot3, ad.plot1, ltdl.fix.df`

Examples

```r
# Make test data available
data(ad.test)
attach(ad.test)

# Plot analytical duplicate analyses as ratios in acquisition sequence
ad.plot4(Cu, ifalt = TRUE)
```
## Description


## Usage

data(ad.test)

## Format

A data frame with 160 observations on the following 2 variables. Unique identifiers are present in the data frame, use `dimnames(ad.test)[[1]]` to access or display them.

- **RS**: the Replicate Status code.
- **Cu**: the copper determinations, mg/kg.

## Details

The Replicate Status code indicates the 'position' of the geochemical sample in the QA/QC structure. RS = 8 indicates analytical duplicate, RS = 2 indicates the field duplicate, and RS = 1 indicates a routine regional coverage site that was 'duplicated'. All other routine regional coverage sites are coded RS = 0. In this file the data record for the analytical duplicate is followed by the data record for the physical sample the duplicate was split from. The analytical duplicate may be split from any field sample, but preferably from one of the field duplicates, permitting a more incisive investigation of sampling and analytical variability using function `gx.triples.aov`.

## Source

Internal Geological Survey of Canada files.
Additive Log-Ratio (alr) transformation

Description

Undertakes an additive log-ratio transformation to remove the effects of closure in a data matrix.

Usage

\text{alr}(xx, j = \text{NULL}, \text{ifclose} = \text{FALSE}, \text{ifwarn} = \text{TRUE})

Arguments

- \text{xx}: a \text{n by p} matrix to be additively log ratioed. It is essential that a single unit of measurement is used. Thus it may be required to convert, for example, determinations in percent to ppm (mg/kg) so that all measurements are in ppm prior to executing this function. Natural logarithms are used.
- \text{j}: the index number of the element in the range [1:p] to be used as the divisor, \text{j}, must be defined, there is no default index.
- \text{ifclose}: if it is required to close a data set prior to transformation set \text{ifclose} = \text{TRUE}.
- \text{ifwarn}: by default \text{ifwarn} = \text{TRUE} which generates a reminder/warning that when carrying out an additive log-ratio transformation all the data must be in the same measurement units. The message can be suppressed by setting \text{ifwarn} = \text{FALSE}.

Details

Most analytical chemical data for major, minor and trace elements are of a closed form, i.e. for a physical individual sample they sum to a constant, whether it be percent, ppm (mg/kg), or some other units. It does not matter that only some components contributing to the constant sum are present in the matrix, the data are closed. As a result, as some elements increase in concentration others must decrease, this leads to correlation measures and graphical presentations that do not reflect the true underlying relationships. An additive log-ratio is one procedure for removing closure effects, others are centred log-ratios (\text{clr}) and isometric log-ratios (\text{ilr}).

Care should be taken in selecting the variable, index = \text{j}, for use as the divisor. Variables lacking sufficient significant figures in their quantification, or variables measured at close to their measurement detection limits, should be avoided.

It is worth noting that when the alr transform is undertaken with a geochemically conservative element selected as the divisor and two elements are then displayed in an x-y plot the result is a Pearce Element Ratio plot (Pearce, 1968) with log scaling.

Value

\text{x}: a \text{n by (p-1)} matrix of additively log-ratioed values, the \text{j}-th column of the matrix being dropped.
Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`. Any rows containing NAs in the data matrix are removed prior to undertaking the transformation.

Author(s)

Robert G. Garrett

References

Aitchison, J. and Egozcue, J.J., 2005. Compositional data analysis; where are we and where should we be heading. Mathematical Geology, 37(7):829-850.

See Also

clr, ilr, ltdl.fix.df, remove.na

Examples

```r
## Make test data available
data(sind)
sind.mat <- as.matrix(sind[, -c(1:3)])

## Undertake alr transform, use Pb [j = 6] as the divisor,
## note necessity of converting percent Fe to mg/kg
sind.mat[, 2] <- sind.mat[, 2] * 10000
temp <- alr(sind.mat, 6)
temp

## Clean-up
rm(sind.mat)
rm(temp)
```
**alts2dups**

Create a Matrix of Duplicate Pairs from Sequential Data

**Description**

Function to take data stored as stacked records or alternating rows of records for `ndup` duplicate pairs and generate a matrix with `ndup` rows and two columns for the duplicate data, for further details see `x` in Arguments below. The function returns a matrix for processing.

**Usage**

`alts2dups(x, ifalt = FALSE)`

**Arguments**

- `x` a column vector from a matrix or data frame, `x[1]`, ..., `x[2×n]`. The default is that the first `n` members of the vector are the first measurements and the second `n` members are the duplicate measurements. If the measurements alternate, i.e. duplicate pair 1 measurement 1 followed by measurement 2, etc., set `ifalt = TRUE`.
- `ifalt` set `ifalt = TRUE` to accommodate alternating sets of paired observations.

**Details**

Function provides an easy procedure with function `plot` to present duplicate data as an x-y plot. The matrix generated can be passed directly to `plot` in its call and the matrix column headings synthesized from the input variable name are used as plot axis titles.

**Value**

`xx` an `ndup` by 2 matrix containing the duplicate pairs. Row names are sequenced `1:ndup` and the two column names are generated from the input variable name, `x` with `.1` or `.2` appended, respectively.

**Note**

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

**Author(s)**

Robert G. Garrett
anova1

**Analysis of Variance (ANOVA)**

**Description**

Undertakes a random effects model Analysis of Variance (ANOVA) on a set of duplicate measurements to determine if the analytical, or combined sampling and analytical, (within) variability is significantly smaller than the variability across the duplicates. For data stored in alternate form use `anovaQ`.

**Usage**

```r
anova1(x1, x2, xname = deparse(substitute(x1)), log = FALSE)
```

**Arguments**

- **x1**: a column vector from a matrix or data frame, \( x1[1], \ldots, x1[n] \).
- **x2**: another column vector from the matrix or data frame, \( x2[1], \ldots, x2[n] \). \( x1 \) and \( x2 \) must be of identical length, \( n \), where \( x2 \) is a duplicate measurement of \( x1 \).
- **xname**: by default the character string for \( x1 \) is used for the title. An alternate title can be displayed with \( xname = "text string" \), see Examples.
- **log**: if a logarithmic transformation (base 10) of the data is required to meet homogeneity of variance considerations (i.e. severe heteroscedasticity) set \( log = TRUE \). This is also advisable if the range of the observations exceeds 1.5 orders of magnitude.

**Examples**

```r
## Make test data available
data(ad.test)
attach(ad.test)

## Save matrix of duplicate pairs
test.save <- alts2dups(Cu, ifalt = TRUE)

## Plot analytical duplicate analyses
plot(alts2dups(Cu, ifalt = TRUE), log = "xy")

## Clean-up and detach test data
rm(test.save)
detach(ad.test)
```
Details

In field geochemical surveys the combined sampling and analytical variability is more important than analytical variability alone. If the at site (within) variability is not significantly smaller than the between duplicate sites variability it cannot be stated that there are statistically significant spatial patterns in the data, and they are likely not suitable for mapping. This may not mean that the data cannot be used to recognize individuals with above threshold or action level observations. However, under these conditions there also may be above threshold or action level instances that the survey data have failed to detect (Garrett, 1983).

A random effects ANOVA is undertaken, the ANOVA table is displayed, together with estimates of the variance components, i.e. how much of the total variability is between and within the duplicate measurements, and the USGS mapping reliability measures of V and Vm (Miesch et al., 1976). Additionally, the data are investigated through a two-way model following the procedure of Bolviken and Sinding-Larsen (1973).

If the data are as a single concatenated vector from a matrix or data frame as \( x[1], \ldots, x[n] \) followed by \( x[n+1], \ldots, x[2n] \), or alternated as \( x[1] \) and \( x[2] \) being a pair through to \( x[2*i+1] \) and \( x[2*i+2] \), for the \( i \) in \( 1:n \) duplicate pairs use function `anova2`.

Note

The script does not follow a standard computation of Mean Squares, but is based on a procedure developed after Garrett (1969) for use in the field in the 1970s when pocket calculators first had mean and standard deviation functions.

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Duplicate pairs \( x_1, x_2 \) containing any NAs are omitted from the calculations.

If a log transformation is undertaken and any less than or equal to zero values occur in the data the function will halt with a warning to that effect.

Author(s)

Robert G. Garrett

References


See Also

anova2, ltdl.fix.df

Examples

```r
## Make test data available
data(ms.data1)
attach(ms.data1)

## Undertake an ANOVA for duplicate measurements on rock samples
anova1(MS.1, MS.2, log = TRUE,
    xname = "Duplicate measurements of Magnetic Susceptibility")

## Detach test data
detach(ms.data1)
```

anova2

Analysis of Variance (ANOVA), Alternate Input

Description

Function to prepare data stored in alternate forms from that expected by function `anova1` for its use. For further details see `x` in Arguments below.

Usage

`anova2(x, xname = deparse(substitute(x)), log = FALSE, ifalt = FALSE)`

Arguments

- `x` a column vector from a matrix or data frame, `x[1], \ldots, x[2*n]`. The default is that the first `n` members of the vector are the first measurements and the second `n` members are the duplicate measurements. If the measurements alternate, i.e. duplicate pair 1 measurement 1 followed by measurement 2, etc., set `ifalt = TRUE`.
- `xname` by default the character string for `x` is used for the title. An alternate title can be displayed with `xname = "text string"`, see Examples.
- `log` if a logarithmic transformation (base 10) of the data is required to meet homogeneity of variance considerations (i.e. severe heteroscedasticity) set `log = TRUE`. This is also advisable if the range of the observations exceeds 1.5 orders of magnitude.
- `ifalt` set `ifalt = TRUE` to accommodate alternating sets of paired observations.

Details

For further details see `anova1`.

If the data are as `n` duplicate pairs, `x1` and `x2`, use function `anova1`.
Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Author(s)

Robert G. Garrett

See Also

`anova`, `ltdl.fix.df`

Examples

```r
## Make test data available
data(ms.data2)
attach(ms.data2)

## Undertake an ANOVA for duplicate measurements on rock samples
anova2(MS, log = TRUE,
xname = "Duplicate measurements of Magnetic Susceptibility")

## Detach test data
detach(ms.data2)

## Make test data available
data(ms.data3)
attach(ms.data3)

## Undertake an ANOVA for duplicate measurements on rock samples
anova2(MS, log = TRUE, ifalt = TRUE,
xname = "Duplicate measurements of Magnetic Susceptibility")

## Detach test data
detach(ms.data3)
```

---

**bwplots**

Plot Vertical Box-and-Whisker Plots

**Description**

Plots a series of vertical box-and-whisker plots (Garrett, 1988) where the individual plots represent the data subdivided by the value of some factor. Optionally the y-axis may be scaled logarithmically (base 10). A variety of other plot options are available, see Details and Note below.
Usage

`bwplots(x, by, log = FALSE, wend = 0.05, notch = TRUE, xlab = "", ylab = deparse(substitute(x)), ylim = NULL, main = "", label = NULL, plot.order = NULL, xpos = NA, width, space = 0.25, las = 1, cex.axis = 1, adj = 0.5, add = FALSE, ssll = 1, colr = 8, pch = 3, ...)`

Arguments

- **x**: name of the variable to be plotted.
- **by**: the name of the factor variable to be used to subdivide the data. See Details below for when by is undefined.
- **log**: if it is required to display the data with logarithmic (y-axis) scaling, set `log = TRUE`.
- **wend**: the locations of the whisker-ends have to be defined. By default these are at the 5th and 95th percentiles of the data. Setting `wend = 0.02` plots the whisker ends at the 2nd and 98th percentiles.
- **notch**: determines if the boxplots are to be “notched” such that the notches indicate the 95% confidence intervals for the medians. The default is to notch the boxplots, to suppress the notches set `notch = FALSE`. See Details below.
- **xlab**: a title for the x-axis, by default none is provided. A title may be provided, see Examples.
- **ylab**: by default the character string for x is used for the y-axis title. An alternate title can be displayed with `ylab = "text string"`, see Examples.
- **ylim**: only for `log = FALSE`, defines the limits of the y-axis if the default limits based on the range of the data are unsatisfactory. It can be used to ensure the y-axis scaling in multiple sets of boxplots are the same to facilitate visual comparison.
- **main**: a main title may be added optionally above the display by setting `main`, e.g., `main = "Kola Project, 1995"`.
- **label**: by default the character strings defining the factors are used to label the boxplots along the x-axis. Alternate labels can be provided with `label = c("Alt1", "Alt2", "Alt3")`, see Examples.
- **plot.order**: provides an alternate order for the boxplots. Thus, `plot.order = c(2, 1, 3)` will plot the 2nd ordered factor in the 1st position, the 1st in the 2nd, and the 3rd in its 3rd ordered position, see Details and Examples below.
- **xpos**: the locations along the x-axis for the individual vertical boxplots to be plotted. By default this is set to `NA`, which causes default equally spaced positions to be used, i.e. boxplot 1 plots at value 1 on the x-axis, boxplot 2 at value 2, etc., up to boxplot “n” at value “n”. See Details below for defining xpos.
- **width**: the width of the boxes, by default this is set to the minimum distance between all adjacent boxplots times the value of space. With the default values of xpos this results in a minimum difference of 1, and with the default of space = 0.25 the width is computed as 0.25. To specify different widths for all boxplots use, for example, `width = c(0.3)`. See Details below for changing individual boxplot widths.
space 

the space between the individual boxplots, by default this is 0.25 x-axis units.

las

controls whether the x-axis labels are written parallel to the x-axis, the default \( \text{las} = 1 \), or are written down from the x-axis by setting \( \text{las} = 2 \). See also, Details below.

cex.axis

cpy controls the size of the font used for the factor labels plotted along the x-axis. By default this is 1, however, if the labels are long it is sometimes necessary to use a smaller font, for example \( \text{cex.axis} = 0.8 \) results in a font 80% of normal size.

adj

cpy controls justification of the x-axis labels. By default they are centred, \( \text{adj} = 0.5 \), to left justify them if the labels are written downwards set \( \text{adj} = 0 \).

add

permits the user to plot additional boxplots into an existing display. It is recommended that this option is left as \( \text{add} = \text{FALSE} \).

ssl,

cpy determines the minimum data subset size for which a subset will be plotted. By default this is set to 1, which leads to only a circle with a median bar being plotted, as the subset size increases additional features of the box-and-whisker plot are displayed. If \( \text{ssl} \) results in subset boxplots not being plotted, a gap is left and the factor label is still plotted on the x-axis.

colr

by default the boxes are infilled in grey, \( \text{colr} = 8 \). If no infill is required, set \( \text{colr} = 0 \). See \text{display.lty} for the range of available colours.

pch

by default the plotting symbol for the subset maxima and minima are set to a plus, \( \text{pch} = 3 \), alternate plotting symbols may be chosen from those displayed by \text{display.marks}.

... further arguments to be passed to methods. For example, the size of the axis titles by setting \( \text{cex.lab} \), and the size of the plot title by setting \( \text{cex.main} \). For example, if it is required to make the plot title smaller, add \( \text{cex.main} = 0.9 \) to reduce the font size by 10%.

Details

There are two ways to execute this function. Firstly by defining \( x \) and \( y \), and secondly by combining the two variables with the \text{split} function. See the first two examples below. The \text{split} function can be useful if the factors to use in the boxplot are to be generated at run-time, see the last example below. Note that when the \text{split} construct is used instead of by the whole \text{split} statement will be displayed as the default y-axis title. Also note that when using by the subsets are listed in the order that the factors are encountered in the data, but when using \text{split} the subsets are listed alphabetically. In either case they can be re-ordered using \text{plot.order}, see Examples.

In a box-and-whisker plot there are two special cases. When \( \text{wew} = 0 \) the whiskers extend to the observed minima and maxima that are not plotted with the plus symbol. When \( \text{wew} = 0.25 \) no whiskers or the data minima and maxima are plotted, only the medians and boxes representing the span of the middle 50% of the data are displayed.

The \text{width} option can be used to define different widths for the individual boxplots. For example, the widths could be scaled to be proportional to the subset population sizes as some function of the square root (\( \text{const} \times \text{sqrt(n)} \)) or logarithm (\( \text{const} \times \text{log10(n)} \)) of those sizes (\( n \)). The constant, \( \text{const} \), would need to be chosen so that on average the width of the individual boxes would be approximately 0.25, see Example below. It may be desirable for cosmetic purposes to adjust the positions of the boxes along the x-axis, this can be achieved by specifying \text{xpos}. 

Long subset (factor) names can lead to display problems, changing the `las` parameter from its default of `las = 1` which plots subset labels parallel to the axis to `las = 2`, to plot perpendicular to the axis, can help. It may also help to use `label` and split the character string into two lines, e.g., by changing the string "Granodiorite" that was supplied to replace the coded factor variable `GRDR` to "Gran-
diorite". If this, or setting `las = 2`, causes a conflict with the x-axis title, if one is needed, the title can be moved down a line by using `xlab = "\nLithological Units"`. In both cases the `\n` forces the following text to be placed on the next lower line.

If there are more than 7 labels (subsets) and no alternate labels are provided `las` is set to 2, otherwise some labels may fail to be displayed.

The notches in the boxplots indicate the 95% confidence intervals for the medians and can extend beyond the upper and lower limits of the boxes indicating the middle 50% of the data when subset population sizes are small. The confidence intervals are estimated using the binomial theorem. It can be argued that for small populations a normal approximation would be better. However, it was decided to remain with a non-parametric estimate to be consistent with the use of non-parametric statistics in this display.

**Note**

This function is based on a script shared by Doug Nychka on S-News, April 28, 1992.

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Any NAs in the data vector are removed prior to preparing the boxplots.

For summary statistics displays to complement the graphics see `gx.summary.groups` or `framework.summary`.

**Author(s)**

Douglas W. Nychka and Robert G. Garrett

**References**


**See Also**

`cat2list, ltdl.fix.df`

**Examples**

```r
## Make test data available
data(kola.c)
attach(kola.c)

## Display a simple box-and-whisker plot
bwplots(Cu, by = COUNTRY)
bwplots(split(Cu,COUNTRY))
```
## bwplots.by.var

### Plot Vertical Box-and-Whisker Plots for Variables

**Description**

Plots a series of vertical box-and-whisker plots (Garrett, 1988) where the individual plots represent the data subdivided by variables. Optionally the y-axis may be scaled logarithmically (base 10). A variety of other plot options are available, see Details and Note below.
Usage

bwplots.by.var(xmat, log = FALSE, wend = 0.05, notch = FALSE,
    xlab = "Measured Variables", ylab = "Reported Values",
    main = "", label = NULL, plot.order = NULL, xpos = NA,
    las = 1, cex.axis = 1, adj = 0.5, colr = 8, pch = 3, ...)

Arguments

xmat
the data matrix or data frame containing the data (variables).

log
if it is required to display the data with logarithmic (y-axis) scaling, set log = TRUE.

wend
the locations of the whisker-ends has to be defined. By default these are at the
5th and 95th percentiles of the data. Setting wend = 0.02 plots the whisker
ends at the 2nd and 98th percentiles. See Details below.

notch
determines if the boxplots are to be “notched” such that the notches indicate
the 95% confidence intervals for the medians. The default is not to notch
the boxplots, to have notches set notch = TRUE.

xlab
a title for the x-axis, by default xlab = "Measured Variables".

ylab
a title for the y-axis, by default ylab = "Reported Values", alternate titling
may be provided, see Examples.

main
a main title may be added optionally above the display by setting main, e.g.,
main = "Kola Project, 1995".

label
by default the character strings defining the variables are used to label the box-
plots along the x-axis. Alternate labels can be provided with
label = c("Alt1", "Alt2", "Alt3"), see Examples.

plot.order
provides an alternate order for the boxplots. By default the boxplots are plotted
in alphabetical order of the factor variables. Thus, plot.order = c(2, 1, 3)
will plot the 2nd alphabetically ordered factor in the 1st position, the 1st in the
2nd, and the 3rd in its alphabetically 3rd ordered position.

xpos
the locations along the x-axis for the individual vertical boxplots to be plotted.
By default this is set to NA, which causes default equally spaced positions to be
used, i.e. boxplot 1 plots at value 1 on the x-axis, boxplot 2 at value 2, etc., up
to boxplot "n" at value "n". See Details below for defining xpos.

las
controls whether the x-axis labels are written parallel to the x-axis, the default
las = 1, or are written down from the x-axis by setting las = 2. See also,
Details below.

cex.axis
controls the size of the font used for the factor labels plotted along the x-axis.
By default this is 1, however, if the labels are long it is sometimes necessary to
use a smaller font, for example cex.axis = 0.8 results in a font 80% of normal
size.

adj
controls justification of the x-axis labels. By default they are centred, adj = 0.5,
to left justify them if the labels are written downwards set adj = 0.

colr
by default the boxes are infilled in grey, colr = 8. If no infill is required, set
colr = 0. See display.lty for the range of available colours.
by default the plotting symbol for the subset maxima and minima are set to a plus, \texttt{pch = 3}, alternate plotting symbols may be chosen from those displayed by \texttt{display.marks}.

... further arguments to be passed to methods. For example, the size of the axis titles by setting \texttt{cex.lab}, and the size of the plot title by setting \texttt{cex.main}. For example, if it is required to make the plot title smaller, add \texttt{cex.main = 0.9} to reduce the font size by 10%.

Details

There are two ways to provide data to this function. Firstly, if all the variables in a data frame are to be displayed, and there are no factor variables, the data frame name can be entered for \texttt{xmat}. However, if there are factor variables, or only a subset of the variables are to be displayed, the data are entered via the \texttt{cbind} construct, see Examples below.

In a box-and-whisker plot there are two special cases. When \texttt{wend = 0} the whiskers extend to the observed minima and maxima that are not plotted with the plus symbol. When \texttt{wend = 0.25} no whiskers or the data minima and maxima are plotted, only the medians and boxes representing the span of the middle 50\% of the data are displayed.

Long variable names can lead to display problems, changing the \texttt{las} parameter from its default of \texttt{las = 1} which plots subset labels parallel to the axis to \texttt{las = 2}, to plot perpendicular to the axis, can help. It may also help to use \texttt{label} and split the character string into two lines, e.g., by changing the string "Specific Conductivity" that was supplied to replace the variable name \texttt{SC} to "Specific\nConductivity". If this, or setting \texttt{las = 2}, causes a conflict with the x-axis title, if one is needed, the title can be moved down a line by using \texttt{xlab = "\nPhysical soil properties"}. In both cases the \texttt{\n} forces the following text to be placed on the next lower line.

If there are more than 7 labels (variables) and no alternate labels are provided \texttt{las} is set to 2, otherwise some variable names may fail to be displayed.

The notches in the boxplots indicate the 95\% confidence intervals for the medians and can extend beyond the upper and lower limits of the boxes indicating the middle 50\% of the data when subset population sizes are small. The confidence intervals are estimated using the binomial theorem. It can be argued that for small populations a normal approximation would be better. However, it was decided to remain with a non-parametric estimate to be consistent with the use of non-parametric statistics in this display.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see \texttt{ltdl.fix.df}.

Any NAs in the data vectors are removed prior to preparing the boxplots.

For a summary statistics display to complement the graphics see \texttt{gx.summary.mat}.

Author(s)

Robert G. Garrett
References

See Also
bwplots, var2fact, ltdl.fix.df

Examples

```r
## Make test data available
data(kola.c)
attach(kola.c)

## Display a simple box-and-whisker plot for measured variables
bwplots.by.var(cbind(Co,Cu,Ni))

## Display a more appropriately labelled and scaled box-and-whisker plot
bwplots.by.var(cbind(Co,Cu,Ni), log = TRUE,
ylab = "Levels (mg/kg) in <2 mm Kola C-horizon soil")

## Detach test data
detach(kola.c)

## Make test data available
data(ms.data1)
attach(ms.data1)

## Display variables in a data frame extending the whiskers to the
## 2nd and 98th percentiles of the data, remembering to omit the
## sample IDs
bwplots.by.var(ms.data1[, -1], log = TRUE, wend = 0.02)

## Detach test data
detach(ms.data1)
```

---

**bxplot**

Plot a Horizontal Boxplot or Box-and-Whisker Plot

**Description**
Plots a single horizontal boxplot as part of the multi-panel display provided by function *shape*, the default is a Tukey boxplot, alternately a box-and-whisker plot (Garrett, 1988) may be displayed. Optionally the x-axis may be scaled logarithmically (base 10).
bxplot

Usage

bxplot(xx, xlab = deparse(substitute(xx)), log = FALSE,
       ifbw = FALSE, wend = 0.05, xlim = NULL, main = "", ifn = TRUE,
       colr = 8, cex = 1, ...)

Arguments

xx name of the variable to be plotted.

xlab by default the character string for xx is used for the x-axis title. An alternate title
       can be displayed with xlab = "text string", see Examples.

log if it is required to display the data with logarithmic (x-axis) scaling, set log = TRUE.

ifbw the default is to plot a horizontal Tukey boxplot, if a box-and-whisker plot is
       required set ifbw = TRUE.

wend if ifbw = TRUE the locations of the whisker-ends have to be defined. By default
       these are at the 5th and 95th percentiles of the data. Setting wend = 0.02 plots
       the whisker ends at the 2nd and 98th percentiles.

xlim when used in the shape function, xlim is determined by gx.hist and used to
       ensure all four panels in shape have the same x-axis scaling. However when
       used stand-alone the limits may be user-defined by setting xlim, see Note below.

main when used stand-alone a title may be added optionally above the plot by setting
       main, e.g., main = "Kola Project, 1995".

ifn an internal ‘switch’ set FALSE to suppress the addition of the sample size to the
       plot.

colr by default the box is infilled in grey, colr = 8. If no infill is required, set
       colr = 0. See display.lty for the range of available colours.

cex by default the size of the text for data set size, N, is set to 80%, i.e. cex = 0.8,
       and may be changed if required.

... further arguments to be passed to methods. For example, the size of the axis
       scale annotation can be change by setting cex.axis, the size of the axis titles by
       setting cex.lab, and the size of the plot title by setting cex.main. For example,
       if it is required to make the plot title smaller, add cex.main = 0.9 to reduce the
       font size by 10%.

Details

The function can be used stand-alone, but as Tukey boxplots and box-and-whisker plots are usu-
ally used to compare the distributions of data subsets the functions tbplots (Tukey boxplots) and
bwplots (box-and-whisker plots) are required for that purpose.

When the boxplot is displayed on a logarithmically scaled x-axis, the data are log transformed prior
to the computation of the positions of the fences used in the Tukey boxplot to identify near and far
outliers, plotted as plusses and circles, respectively.

In a box-and-whisker plot there are two special cases. When wend = 0 the whiskers extend to the
observed minima and maxima that are not plotted with the plus symbol. When wend = 0.25 no
whiskers or the data minimum and maximum are plotted, only the median and box representing the
span of the middle 50 percent of the data are displayed.
Note

Any less than detection limit values represented by negative values, or zeros or numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`. Any NAs in the data vector are removed prior to displaying the plot.

If the default selection for `xlim` is inappropriate it can be set, e.g., `xlim = c(0, 200)` or `c(2, 200)`, the latter being appropriate for a logarithmically scaled plot, i.e. `log = TRUE`. If the defined limits lie within the observed data range a truncated plot will be displayed. If this occurs the number of data points omitted is displayed below the total number of observations.

If it is desired to prepare a display of data falling within a defined part of the actual data range, then either a data subset can be prepared externally using the appropriate R syntax, or `xx` may be defined in the function call as, for example, `Cu[Cu < some.value]` which would remove the influence of one or more outliers having values greater than `some.value`. In this case the number of data values displayed will be the number that are `<some.value`.

Author(s)

Robert G. Garrett

References


See Also

`shape, display.lty, ltdl.fix.df, remove.na`

Examples

```r
## Make test data available
data(kola.o)
attach(kola.o)

## Display a simple boxplot
bxplot(Cu)

## Display a more appropriately labelled and scaled boxplot
bxplot(Cu, xlab = "Cu (mg/kg) in <2 mm Kola O-horizon soil", log = TRUE)

## Display a box-and-whisker plot with whiskers ending at the 2nd and
## 98th percentiles
bxplot(Cu, xlab = "Cu (mg/kg) in <2 mm Kola O-horizon soil", ifbw = TRUE,
wend = 0.02, log = TRUE)

## Detach test data
detach(kola.o)
```
Prepare a Concentration-Area (C-A) Plot

Description

Displays a concentration-area (C-A) plot to assess whether the data are spatially multi-fractal (Cheng et al., 1994; Cheng and Agterberg, 1995) as a part of a four panel display. This procedure is useful for determining if multiple populations that are spatially dependent are present in a data set. It can be used to determine the practical limits, upper or lower bounds, of the influence of the biogeochemical processes behind the spatial distribution of the data. Optionally the data may be logarithmically transformed (base 10) prior to interpolation, the points may be 'jittered' (see Arguments below), the size of the interpolated grid may be modified, and alternate colour schemes can be chosen for display of the interpolated data.

Usage

```r
caplot(x, y, z, zname = deparse(substitute(z)),
caname = deparse(substitute(z)), log = TRUE, ifjit = FALSE,
ifrev = FALSE, ngrid = 100, colr = topo.colors(16),
 xlink = "Easting", ycoord = "Northing")
```

Arguments

- `x` - name of the x-axis spatial coordinate, the eastings.
- `y` - name of the y-axis spatial coordinate, the northings.
- `z` - name of the variable to be processed and plotted.
- `zname` - by default the character string for `z` is used for the titles of the x-axes of the CPP (Cumulative Normal Percentage Probability) and C-A plot panels. An alternate title can be displayed with `zname = "text string"`, see Examples.
- `caname` - a title for the image of the interpolated data. It is often desirable to replace the default title of the input variable name text string with a more informative title, e.g., `caname = "Kola Project, 1995\nCu (mg/kg) in <2 mm O-horizon soil"`. For no title, set `caname = ""`.
- `log` - the default is set to `log = TRUE` as in most cases this function is used with data from a ‘closed’ composition or are positively skewed, where a logarithmic data transform is appropriate. If it is required to undertake the C-A plot interpolation without a logarithmic data transformation, set `log = FALSE`. This also results in the accompanying probability (CPP) plots being arithmically scaled (x-axes).
- `ifjit` - if there is a possibility that the data set contains multiple measurements at an identical spatial (x,y) location set `ifjit = TRUE`. The presence of multiple data at an identical location will cause the Akima (1996) interpolation function to fail.
- `ifrev` - by default the empirical C-A function is plotted from highest value to lowest, `ifrev = FALSE`. As the C-A plot is a log-log display this provides greater
detail for the highest values. The direction of accumulation can be key in detecting multi-fractal patterns, it is usually informative to also prepare a plot with \texttt{ifrev = TRUE}, i.e. accumulation from lowest to highest values. To see a dramatic example of this, run the Examples below.

\begin{itemize}
  \item \texttt{ngrid} by default \texttt{ngrid = 100}, this results in the data being interpolated into a 100 x 100 grid that extends between the data set’s spatial extremes determined for the \((x,y)\) spatial coordinates for the data. See Details below.
  \item \texttt{colr} by default the \texttt{topo.colors(16)} palette is used to render the interpolated grid as an image. For alternative palettes see \texttt{colors}, and see Details below.
  \item \texttt{xcoord} a title for the x-axis, defaults to \texttt{Easting}.
  \item \texttt{ycoord} a title for the y-axis, defaults to \texttt{Northing}.
\end{itemize}

\textbf{Details}

The function creates a four panel display. The percentage cumulative probability (CPP) plot of the data in the upper left, and the CPP plot of the interpolated data to be used in the C-A plot in the upper right. The lower left panel contains an image of the interpolated data, and the lower right the C-A plot.

Akima’s (1978, 1996) interpolation function is used to obtain a linear interpolation between the spatial data values. If the data are from a ‘closed’ composition or are positively skewed the use of a logarithmic data transformation, \texttt{log = TRUE}, is highly recommended, as noted above this is commonly the case and is the default. Following generation of the interpolated grid and prior to further processing the interpolated grid values are clipped by the convex-hull of the spatial locations, therefore there is no interpolation beyond the spatial extent, support, of the data is displayed.

The use of the \texttt{topo.colors(16)} palette to display the image of the interpolated values leads to low values being plotted in blue, and as the interpolated values increase they take on green, yellow and orange colors. For a grey-scale display for black-and-white use set \texttt{colr = grey(0:8/8)}. This leads to lowest interpolated values being plotted in black and the highest in white, using \texttt{colr = grey(8:0/8)} reverses this, with the lowest values being plotted in white and the highest in black. In either case, if the values plotted in white occur at the study area boundary, i.e. at the convex hull, the difference between no data and white cannot be discerned.

For preparation of the C-A plot the ordered vector of interpolated values is used as a surrogate for the measurement of area greater than, or less than, a stated interpolated value. The cumulative percentage count of the interpolated values being plotted on the y-axis of the C-A plot. As noted above, it is both informative and important to display the C-A plot accumulated both upwards and downwards.

\textbf{Note}

This wrapper function was developed from a S-Plus function to prepare C-A plots using Akima’s (1978, 1996) interpolation procedure written by Graeme Bonham-Carter, Geological Survey of Canada, in April 2004.

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see \texttt{ltdl.fix.df}.

Any sites with NAs in their \((x,y,z)\) data are removed prior to spatial interpolation and preparation of the C-A plot.
In some R installations the generation of multi-panel displays and the use of function eqscplot from package MASS causes warning messages related to graphics parameters to be displayed on the current device. These may be suppressed by entering `options(warn = -1)` on the R command line, or that line may be included in a first function prepared by the user for loading the ‘rgr’ package, etc.

**Author(s)**

Graham F. Bonham-Carter and Robert G. Garrett

**References**


**See Also**

cnpplt, interp, colors, ltdl.fix.df

**Examples**

```r
## The following examples are commented out as package akima is not
## automatically made available as it is only a suggest, not a depends,
## and therefore caplot fails when the examples are run during package
## checking and building

## Make test data available
## data(kola.o)
## attach(kola.o)

## A default (uninformative) C-A plot
## caplot(UTME/1000, UTMN/1000, Cu)

## Plot a more appropriately scaled (log transformed data) and
## titled display
## caplot(UTME/1000, UTMN/1000, Cu, log = TRUE,
## zname = "Cu (mg/kg) in\n<2 mm O-horizon soil",
## caname = "Kola Project, 1995\nCu (mg/kg) in <2 mm O-horizon soil")

## Plot as above but with the C-A plot accumulation reversed
## caplot(UTME/1000, UTMN/1000, Cu, log = TRUE, ifrev = TRUE,
## zname = "Cu (mg/kg) in\n<2 mm O-horizon soil",
## caname = "Kola Project, 1995\nCu (mg/kg) in <2 mm O-horizon soil")
```
```r
## Detach test data
## detach(kola.o)
```

---

**cat2list** *Divides Data into Subsets by Factor*

**Description**

Converts data into a list form where data are grouped together by factor. Achieves the same objective as the base function `split`.

**Usage**

```r
cat2list(x, a)
```

**Arguments**

- `x`: name of the data variable to be processed.
- `a`: name of the factor variable by which the data are to be split.

**Value**

- `data`: a list containing factors as columns and the values for those factors as rows. The order of the resulting groups, subsets, is the order in which the factor variable names were encountered in parameter `a` passed to the function.

**Note**

This function is called by functions `tbplots` and `bwplots` to prepare Tukey boxplots and box-and-whisker plots, respectively. It is an integral part of the script shared by Doug Nychka on S-News, April 28, 1992. As such it may pre-date the time that `split` was added to the S-Plus library.

If `by` is undefined in the calling functions, `tbplots` and `bwplots`, the same result may be achieved by using the `split(x, a)` construct instead of stating `x` as the variable to be displayed as boxplots. In which case the data are grouped, subsetted, in alphabetical order of factor variable names.

**Author(s)**

Douglas W. Nychka


**clr**

*Centred Log-Ratio (clr) transformation*

**Description**

Undertakes a centred log-ratio transformation to remove the effects of closure in a data matrix.

**Usage**

`clr(xx, ifclose = FALSE, ifwarn = TRUE)`

**Arguments**

- `xx`: a `n` by `p` matrix to be log centred. It is essential that a single unit of measurement is used. Thus it may be required to convert, for example, determinations in percent to ppm (mg/kg) so that all measurements are in ppm prior to executing this function. Natural logarithms are used.
- `ifclose`: if it is required to close a data set prior to transformation set `ifclose = TRUE`.
- `ifwarn`: by default `ifwarn = TRUE` which generates a reminder/warning that when carrying out a centred log-ratio transformation all the data must be in the same measurement units. The message can be suppressed by setting `ifwarn = FALSE`.

**Details**

Most analytical chemical data for major, minor and trace elements are of a closed form, i.e. for a physical individual sample they sum to a constant, whether it be percent, ppm (mg/kg), or some other units. It does not matter that only some components contributing to the constant sum are present in the matrix, the data are closed. As a result, as some elements increase in concentration others must decrease, this leads to correlation measures and graphical presentations that do not reflect the true underlying relationships. A centred log-ratio is one procedure for removing closure effects, others are additive log-ratios (`alr`) and isometric log-ratios (`ilr`).

**Value**

- `x`: a `n` by `p` matrix of log-centred values.

**Note**

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Any rows containing NAs in the data matrix are removed prior to undertaking the transformation.

The `clr` transform is suitable for the study of correlation coefficients and subsequent multivariate data analyses. However, for the calculation of Mahalanobis distances, which require matrix inversion, `ilr` should be used. Furthermore, in some cases it is preferable to use an `ilr` transform prior to undertaking a Principal Component or Factor Analysis, however, a `clr` transform is often sufficient.
The ifclose option can be useful if a petrochemical ternary system is under investigation. A data subset for a ternary system may be closed and transformed for investigation in x-y plots and comparison with the inferences that may be drawn from a classical ternary diagram display. Ternary plots are not included in this release of 'rgr', their use is discouraged as they do not reveal the true inter-component relationships. However, their use as classification tools is acknowledged where a user's data may be compared to data for known rock types and processes, etc. R users interested in ternary and classification diagrams rather than exploratory data analysis should investigate GCDkit (ver 2.3, R 2.7.0 2008/05/11) by Janousek, Farrow, Erban and Smid. See also Janousek et al. (2006).

Author(s)

Robert G. Garrett

References

Aitchison, J. and Egozcue, J.J., 2005. Compositional data analysis; where are we and where should we be heading. Mathematical Geology, 37(7):829-850.

See Also

clr, ilr, ltdl.fix.df, remove.na

Examples

```r
## Make test data available
data(sind)
sind.mat <- as.matrix(sind[, -c(1:3)])

## Undertake clr transform, note necessity
## of converting percent Fe to mg/kg
sind.mat[, 2] <- sind.mat[, 2] * 10000
temp <- clr(sind.mat)
temp

## Clean-up and detach test data
rm(sind.mat)
rm(temp)
```
**cnpplt**  
*Cumulative Normal Percentage Probability (CPP) Plot*

**Description**

Displays a cumulative normal percentage probability (CPP) plot, equivalent to a Q-Q plot, as has been traditionally used by physical scientists and engineers.

**Usage**

```r
cnpplt(xx, xlab = deparse(substitute(xx)), ylab = "% Cumulative Probability", log = FALSE, xlim = NULL, main = "", ifqs = FALSE, ifshape = FALSE, pch = 3, cex = 0.8, cexp = 1, cex.axis = 0.8, ...)
```

**Arguments**

- **xx**
  - name of the variable to be plotted.

- **xlab**
  - by default the character string for `xx` is used for the x-axis title. An alternate title can be displayed with `xlab = "text string"`, see Examples.

- **ylab**
  - a title for the y-axis, defaults to "% Cumulative Probability".

- **log**
  - if it is required to display the data with logarithmic (x-axis) scaling, set `log = TRUE`.

- **xlim**
  - when used in the `shape` function, `xlim` is determined by function `gx.hist` and used to ensure all four panels in `shape` have the same x-axis scaling. However, when used stand-alone the limits may be user-defined by setting `xlim`, see Details below.

- **main**
  - when used stand-alone a title may be added optionally above the plot by setting `main`, e.g., `main = "Kola Ecogeochemistry Project, 1995"`.

- **ifqs**
  - setting `ifqs = TRUE` results in horizontal and vertical dotted lines being plotted at the three central quartiles and their values, respectively.

- **ifshape**
  - when used with function `shape` or `caplot` to plot into a panel set `ifshape = TRUE` to ensure only essential probability scale axis labels are displayed to avoid overplotting on the reduced size panel plot.

- **pch**
  - by default the plotting symbol is set to a plus, `pch = 3`, alternate plotting symbols may be chosen from those displayed by `display.marks`, see also `Note` below.

- **cex**
  - by default the size of the text for data set size, `N`, is set to 80%, i.e. `cex = 0.8`, and may be changed if required.

- **cexp**
  - by default the size of the plotting symbol, `pch`, is set to 100%, and may be changed if required.

- **cex.axis**
  - if overplotting occurs in the y-axis labelling the size of the y-axis labels may be reduced by setting `cex.axis` to a number smaller than the default of `cex.axis = 0.8`.
... further arguments to be passed to methods. For example, the size of the axis
titles by setting `cex.lab`, and the size of the plot title by setting `cex.main`. For
example, if it is required to make the plot title smaller, add `cex.main = 0.9` to
reduce the font size by 10%. The colour of the plotting symbols may be changed
from the default black, e.e., `col = 2` for red.

Details

If the default selection for `xlim` is inappropriate it can be set, e.g., `xlim = c(0, 200)` or `c(2, 200)`,
the latter being appropriate for a logarithmically scaled plot, i.e. `log = TRUE`. If the defined limits
lie within the observed data range a truncated plot will be displayed. If this occurs the number of
data points omitted is displayed below the total number of observations.

If it is desired to prepare a display of data falling within a defined part of the actual data range, then
either a data subset can be prepared externally using the appropriate R syntax, or `xx` may be defined
in the function call as, for example, `Cu[Cu < some.value]` which would remove the influence of
one or more outliers having values greater than `some.value`. In this case the number of data values
displayed will be the number that are `<some.value`.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes
representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Any NAs in the data vector are removed prior to displaying the plot.

The available symbols are:

- `pch`: 0 = square, 1 = circle, 2 = triangle, 3 = plus, 4 = X,
  5 = diamond, 6 = upside-down triangle, 7 = square with X,
  8 = asterisk, 9 = diamond with plus, 10 = circle with plus,
  11 = double triangles, 12 = square with plus,
  13 = circle with X, 14 = square with upside-down triangle.

Symbols 15 to 18 are solid in the colour specified:

- 15 = square, 16 = circle, 17 = triangle, 18 = diamond.

Author(s)

Robert G. Garrett

See Also

`display.marks, ltdl.fix.df, remove.na`

Examples

```r
## Make test data available
data(kola.o)
attach(kola.o)

## A stand-alone cumulative normal percentage probability plot
cnpplt(Cu)
```
## A more appropriately labelled and scaled cumulative normal percentage

## probability plot using a cross/x rather than a plus

```r
cnpplt(Cu, xlab = "Cu (mg/kg) in <2 mm O-horizon soil", log = TRUE, pch = 4)
```

## Detach test data

```r
detach(kola.o)
```

### Plot Results of Control Reference Material (CRM) Analyses

#### Description

Function to plot the results of Control Reference material (CRM) analyses in the order in which they occur in a file, assuming that this order is a time-series, so that the presence of ‘drift’ may be recognized, in addition to the presence of gross outliers reflecting ‘analytical problems’. The data are plotted as either values, when the associated standard deviation of the CRM is provided, or percent absolute relative difference from the ‘recommended’ value when a target tolerance level is provided (see below). The expected ‘recommended’ value (long-term mean) for the CRM being displayed must be supplied, together with its associated standard deviation, or a target tolerance level for percent absolute relative difference. By default the CRM recommended value and standard deviation are used to plot red dashed lines at the recommended value +/- 2 standard deviations, and a green line for the recommended value, alternate standard deviation multiples may be provided. This display is also known as a Shewart plot.

#### Usage

```r
crm.plot(xx, xname = deparse(substitute(xx)), crm.mean = NULL, crm.sd = NULL, n.sd = 2, crm.tol = NULL, ...)
```

#### Arguments

- `xx`: a column vector of determinations from a data frame or matrix for a measured parameter on a CRM.
- `xname`: a title can be displayed with the plot and results, e.g., `xname = "Cu (mg/kg)"`. If this field is undefined the character string for `xx` is used as a default.
- `crm.mean`: the recommended value for the CRM. A value must be provided, otherwise the function will terminate.
- `crm.sd`: the standard deviation associated with the recommended value for the CRM. Appropriate red dotted control lines are plotted above and below the mean.
- `n.sd`: by default 2 standard deviation limits are used on the Shewart plot, alternate values may be supplied.
- `crm.tol`: optionally a percentage tolerance level may be provided for the maximum acceptable absolute relative percent difference from the CRM recommended value, in which case a red dotted control line is added to the plot.
- `...`: any additional arguments to be passed to the `plot` function for titling, etc.
Details

Either a standard deviation for the CRM analyses or an upper limit tolerance level must be provided, otherwise the function will fail. If both are provided an percentage absolute relative difference plot is displayed.

Where the input data file contains determinations for more than one CRM, either a subset for the CRM of interest must be created, e.g., with `gx.subset`, or the R construct `Cu[CRM="X"]` must be used to pass the data to the function.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Author(s)

Robert G. Garrett

See Also

`ltdl.fix.df`

Examples

```r
## Make test data available
data(crm.test)
attach(crm.test)

## Plot CRM analyses time-series for CRM-X using the CRM SD estimate
## and the default 2 SD tolerance bounds
crm.plot(Cu[CRM="X"], "Cu(mg/g) in CRM-X", crm.mean = 34.5, crm.sd = 2.19)

crm.plot(Cu[CRM="X"], "Cu(mg/g) in CRM-X", crm.mean = 34.5, crm.sd = 2.19, n.sd = 2.5)

crm.plot(Cu[CRM="X"], "Cu(mg/g) in CRM-X", crm.mean = 34.5, crm.tol = 15)

detach(crm.test)
```
**crm.plot.new**  
*Plot Results of Control Reference Material (CRM) Analyses Adding New Data*

---

**Description**

Function to plot new CRM data with historic CRM data. Summary statistics are estimated for the historic data, and if an established mean and SD are not provided these are used to estimate the tolerance bounds for the new data to be plotted. The user may define the number of SDs for the tolerance bounds, the default is two. The user must provide the number of new CRM determinations to be plotted so that the x-axis may be extended appropriately. The inclusive range of the new data to be plotted must be provided to ensure all the data and tolerance bounds are plotted. The new data are plotted into the display once the function has executed using the R `plot` command, see Examples.

**Usage**

```r
crm.plot.new(xx, xname = deparse(substitute(xx)), crm.mean = NULL,  
crm.sd = NULL, n.sd = 2, crm.new = 0, ylim = NULL, ...)
```

**Arguments**

- **xx**: a column vector of determinations from a data frame or matrix for a measured parameter on a CRM.
- **xname**: a title can be displayed with the plot and results, e.g., xname = "Cu (mg/kg)". If this field is undefined the character string for xx is used as a default.
- **crm.mean**: the recommended value for the CRM. If not defined the mean will be estimated from the input (historic) data.
- **crm.sd**: the standard deviation associated with the recommended value for the CRM. If not defined the SD will be estimated from the input(historic) data. Appropriate red dotted control lines are plotted above and below the mean.
- **n.sd**: by default 2 standard deviation limits are used on the Shewart plot, alternate values may be supplied.
- **crm.new**: the number of new CRM determinations to be added to the plot of historic data. The new data may be in several subsets, crm.new has to be the total of all new determinations to be added.
- **ylim**: the inclusive range of the new determinations to be added. The limits for the y-axis will be determined internally taking into account the limits of the historic data and the estimated 2 standard deviation limits.
- **...**: any additional arguments to be passed to the plot function for titling, etc.
Details

A mean and standard deviation for the historic CRM analyses should be provided. If they are not, estimates are made from the input (historic) data.

Following the completion of the plot and the addition of the new data, the plot may be annotated using the R text function. In the example below, to the right of the cursor position, in red and at 80% font size:

```
> text(locator(1), "New determinations as red asterisks", adj = 0, col = 2, cex = 0.8)
```

If `crm.new` is undefined the historic data will be plotted using either the external, or internal, estimates of the historic mean and standard deviation.

Where the input data file contains determinations for more than one CRM, either a subset for the CRM of interest must be created, e.g., with `gx.subset`, or the R construct `Cu[CRM=="X"]` must be used to pass the data to the function.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Author(s)

Robert G. Garrett

See Also

`crm.plot`, `ltdl.fix.df`

Examples

```r
## Make test data available
data(crm.test)
attach(crm.test)
data(crm.test.new)

## Determine the range of the new determinations to be plotted
range(crm.test.new$Cu)

## Plot CRM analyses time-series for 25 CRM-X determinations using
## a historic CRM mean and SD estimate, and the default 2 SD
## tolerance bounds
crm.plot.new(Cu[CRM=="X"], "Cu(mg/g) in CRM-X", crm.mean = 34.5, crm.sd = 2.19,
crm.new = 10, ylim = c(29, 39))

## Add the 10 new CRM-X determinations to the plot as red asterisks
points(seq(26,35), crm.test.new$Cu, pch = 8, col = 2)

## Plot CRM analyses time-series for 25 CRM-X determinations and
## base the CRM mean, SD estimate and confidence bounds on those
data, using the default 2 SD tolerance bounds

## crm.plot.new(Cu[CRM=="X"], "Cu(mg/g) in CRM-X", crm.new = 10, ylim = c(29, 39))
```
## Description


## Usage

```r
data(crm.test)
```

## Format

A data frame with 97 observations on the following 2 variables. Unique identifiers are present in the data frame, use `dimnames(crm.test)[[1]]` to access or display them.

- **CRM** a code indicating the particular CRM analysed. A factor variable with levels: STSD-1, STSD-2, STSD-3, STSD-4, TILL-4, X, Y, and Z.
- **Cu** the copper determinations, mg/kg.

## Details

The ‘value’ of CRM is used to select the CRM data to be displayed, either by creating a specific subset, e.g., using `gx.subset`, or using the R construct `Cu[CRM="X"]` in the call to function `crm.plot`.

## Source

Internal Geological Survey of Canada files.
Description

A subset, N = 10, of simulated Control Reference Material (CRM) data for copper with mean 34.5 and SD 2.19.

Usage

data(crm.test.new)

Format

A data frame with 10 observations on the following 2 variables. Unique identifiers are present in the data frame, use dimnames(crm.test)[[1]] to access or display them.

- **CRM**: a code indicating the particular CRM simulated. A factor variable with a single level: x.
- **Cu**: the simulated copper determinations, mg/kg.

Details

A set of simulated copper data for demonstrating function `crm.plot.new`, for details of usage see `crm.plot.new`.

Source

`rnorm(10, 34.5, 2.19)`

cutter

Function to Identify in Which Interval a Value Falls

Description

Function to identify in which interval of a set of cut points, `cuts`, a value `x` falls within or beyond. The number of intervals is equal to the number of cut points plus 1. Values of `x` have to exceed the value of the cut point to be allocated to the higher interval.

Usage

cutter(x, cuts)

Arguments

- `x` name of the vector to be processed.
- `cuts` the vector of cut points.
df.test

Value

**xi**  
a vector of the same length as `x` containing an integer between 1 and the number of cut points plus 1 indicating in which interval each value of `x` fell. Values `<cut[1]` have `xi` set to 1, and values `>cut[highest]` have `xi` set to the number of cut points plus 1.

Author(s)

Robert G. Garrett

Examples

```r
## Make test data available
data(kola.c)
attach(kola.c)

## Cut the data into quartiles
xi <- cutter(Cu, quantile(Cu, probs = c(0.25, 0.5, 0.75)))

## Detach test data
detach(kola.c)
```

---

**df.test**  
*Check for the Existence of a Data Frame*

Description

A utility function to determine if a data frame is attached, or exists in the working directory. If the data frame exists the names of the variables are displayed together with the data frame dimensions.

Usage

```r
df.test(dfname)
```

Arguments

- `dfname` name of a data frame.

Details

Based on a function shared on S-News.

Author(s)

Unknown

See Also

`search`, `ls`
Examples

```r
# Make test data available
data(kola.o)

# Check that the data frame kola.o is available
df.test(kola.o)
```

display.ascii.o  

*Display the Windows Latin 1 Font Octal Table*

**Description**

A utility function to display the octal numbers corresponding to the Windows Latin 1 Font.

**Usage**

```r
display.ascii.o()
```

**Details**

The ASCII octal ‘escape codes’ are used to insert special characters in text strings for axis labelling, and titles etc., in graphical displays. For example the escape string `\265` results in the Greek letter mu being displayed.

Based on a function shared on S-News.

**Author(s)**

Unknown

**Examples**

```r
display.ascii.o()
```

display.lty  

*Display Available Line Styles and Colour Codes*

**Description**

Displays the line styles and colours corresponding to `lty = 1 to 9` and `colr = 1 to 9`, respectively.

**Usage**

```r
display.lty()
```
Note

All ‘rgr’ functions that plot boxes or polygons have their default infill colour, colr, set to grey, colr = 8. This may be changed to an alternate colour, colr = 1 to 7 or 9, for no infill colour, set colr = 0.

Author(s)

Robert G. Garrett

display.marks  
Display Available Plotting Marks

Description

Displays the available plotting marks. Where specified, the ‘rgr’ functions use a plus sign, pch = 3, as the plotting symbol, alternate plotting marks may be selected from this display. For example, pch = 1 results in an open circle, the R default, and pch = 4 results in a cross, ‘x’. For additional symbols available only in R (pch = 19:25) see points.

Usage

display.marks()

Note


Author(s)

Various, see Note

display.rainbow  
Display the Colours of the Rainbow(36) Pallette

Description

Displays the available colours in the rainbow(36) pallette to support the selection of alternate colour schemes.

Usage

display.rainbow()

Author(s)

Robert G. Garrett
expit

Inverse-logit transformation(s)

Description

Undertakes an inverse-logit transformation for a vector or single value.

Usage

expit(z)

Arguments

z  
the value(s) to be inverse-logit transformed. Natural logarithms are used.

Details

Most analytical chemical data for major, minor and trace elements are of a closed form, i.e. for a sample they sum to a constant, whether it be percent, ppm (mg/kg), or some other units. It does not matter that only some components contributing to the constant sum are present in the matrix, the data are closed. As a result, as some elements increase in concentration others must decrease, this leads to statistics and graphical presentations that do not reflect the true underlying situation even in situations of univariate data analysis and display. The logit transformation provides an appropriate transformation for univariate compositional data. Procedures for removing closure effects for multivariate data are additive log-ratios (alr), centred log-ratios (clr), and isometric log-ratios (ilr).

Value

p  
the proportion(s) corresponding to the logit transformed value(s), z, passed to the function.

Note

This function is provided so that summary statistics generated by ‘rgr’ functions can be back-transformed to the original units following computations using logit transformed data, see logit.

Author(s)

Robert G. Garrett

References

fences

See Also

logit, alr, clr, ilr

Examples

```r
## Generate test data
z <- c(1.6, 0, -2.3)

## Undertake and display inverse-logit transformation(s)
p <- expit(z)
p

## Clean-up
rm(z)
rm(p)
```

## Generate and Display Fence Values

Description

Function to generate fence values to support the selection of the upper and lower bounds of background variability, i.e., threshold(s) or action levels, when an obvious graphical solution is not visually recognizable.

Usage

```r
fences(xx, units = "ppm", display = TRUE)
```

Arguments

- `xx`: name of the variable to be processed.
- `units`: the units of measurement, options are: "pct", "ppm", "ppb", "ppt". The default is "ppm".
- `display`: the default is to display the tabular output on the current device, i.e., `display = TRUE`. However, when the function is used by `fences.summary` and in order to suppress output to the current device `display = FALSE` as the displayed results will be saved to a text file for subsequent use/editing and reference.

Details

The fence values are computed by several procedures both with and without a logarithmic data transformation and with a logit transformation, together with the 98th percentile of the data for display. Fences are computed following Tukey's boxplot procedure, as median +/- 2 * MAD (Median Absolute Deviation), and mean +/- 2 * SD (Standard Deviation), see Reimann et al. (2005). It is essential that these estimates be viewed in the context of the graphical distributional displays, e.g., `shape` and its graphical components, `gx.hist`, `gx.ecdf`, `cnpplt` and `bxplot`, and if spatial coordinates for the
sample sites are available \texttt{map.eda7}, \texttt{map.eda8} and \texttt{caplot}. The final selection of a range for background or the selection of a threshold level needs to take the statistical and spatial distributions of the data into account. It is also necessary to be aware that it might be appropriate to have more than one background range/threshold in a study or survey (Reimann and Garrett, 2005). The presence of relevant information in the data frame may permit the data to be subset on the basis of that information for display with the \texttt{tbplots}, \texttt{bwplots} and \texttt{gx.cnpplts} functions. If these indicate that the medians and middle 50\%s of the data are visibly different, multiple background ranges may be advisable.

\textbf{Note}

The logit transformation requires that the input value be in the range zero to one. This transformation takes into consideration the closed, constant sum, nature of geochemical analytical data (Filzmoser et al., 2009). Therefore the measurement units must be defined so that the the value can be divided by the appropriate constant. The default is “ppm”, and other acceptable units are “pct”, “ppb” and “ppt”. However, it should be noted that at trace element levels the differences between fences computed with logarithmic and logit transformations are small, and in most applied geochemical applications the logarithmic transformation will suffice. This is not the case for concentrations at major element levels, where the data are more ‘normally’ distributed and fences will be markedly different between untransformed and logit based estimates.

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see \texttt{ltdl.fix.df}. Any NAs in the data vector are removed prior to computing the fences.

\textbf{Author(s)}

Robert G. Garrett

\textbf{References}


\textbf{See Also}

\texttt{fences.summary}, \texttt{ltdl.fix.df}, \texttt{remove.na}, \texttt{logit}, \texttt{expit}

\textbf{Examples}

```r
## Make test data available
data(kola.o)
```
attach(kola.o)

## Display the fences computed for Cu
fences(Cu)

## Detach test data
detach(kola.o)

---

### Generate and Save Fence Values for Data Subsets

#### Description

Function to generate fences and save the values in the R working directory for subsets of the data for a variable when the data can be subdivided by some criterion (factor) such as EcoRegion, Province, physical sample parent material, etc. The function supports the selection of the upper and lower bounds of background variability, and threshold(s) or action levels, when obvious graphical solutions are not visually recognizable.

#### Usage

fences.summary(group, x, file = NULL, units = "ppm")

#### Arguments

- **group**: the name of the factor variable by which the data are to be subset.
- **x**: name of the variable to be processed.
- **file**: the first part of the file name identifying the data source for saving the function output in the R working directory, see Details below.
- **units**: the units of measurement, options are: “pct”, “ppm”, “ppb”, “ppt”. The default is “ppm”.

#### Details

The fence values are computed by several procedures both with and without a logarithmic data transformation and with a logistic transformation, together with the 98th percentile of the data for display. These computations are based on results returned from function `gx.stats`. Fences are computed following Tukey’s boxplot procedure, as median +/- 2 * MAD (Median Absolute Deviation), and mean +/- 2 * SD (Standard Deviation), see Reimann et al. (2005). It is essential that these estimates are viewed in the context of the graphical distributional displays, e.g., `shape` and its graphical components, `gx.hist`, `gx.ecdf`, `cnpplt` and `bxplot`, and if spatial coordinates for the sample sites are available `map.eda7`, `map.eda8` and `caplot`. The final selection of a range for background or the selection of a threshold level needs to take the statistical and spatial distributions of the data into account. It is also necessary to be aware that it might be appropriate to have more than one background range/threshold in an area (Reimann and Garrett, 2005). The presence of relevant information in the data frame may permit the data to be subset on the basis of that information for
display with the `tbplots`, `bwplots` and `gx.cpplts` functions. If these indicate that the medians and middle 50% of the data are visibly different, multiple background ranges may be advisable.

A default file name is generated by concatenating the data frame, group and variable, x, names, separated by ‘.’s and terminating in `.fences.txt`. If file contains text it is used as the first part of the file name identifying the data source for the file to be saved in the specified folder, for example, `file = "D://R_work//Project3//C_soils"`. If no folder is specified the file is saved in the R working directory.

Output to the current device is suppressed. The output file is formatted as a tab delimited file to be read with a spread sheet program. It can be inspected with a text viewer, and column spacings edited for cosmetic purposes with an ASCII editor of the user’s choice.

**Note**

The logit transformation requires that the input value be in the range zero to one. This transformation takes into consideration the closed, constant sum, nature of geochemical analytical data (Filzmoser et al., 2009). Therefore the measurement units must be defined so that the the value can be divided by the appropriate constant. The default is “ppm”, and other acceptable units are “pct”, “ppb” and “ppt”. However, it should be noted that at trace element levels the differences between fences computed with logarithmic and logit transformations are small, and in most applied geochemical applications the logarithmic transformation will suffice. This is not the case for concentrations at major element levels, where the data are more ‘normally’ distributed and fences will be markedly different between untransformed and logit based estimates.

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Any NAs in the data vector are removed prior to computing the fences.

The function `fences` is employed to compute the statistical fence estimates.

**Author(s)**

Robert G. Garrett

**References**


**See Also**

`fences, ltdl.fix.df, remove.na`
Examples

```r
## Make test data available
data(kola.c)
attach(kola.c)

## Saves the file kola_c_COUNTRY_Cu_fences.txt for later use
## in the R working directory.
fences.summary(COUNTRY, Cu, file = "Kola_c_horizon")

## Detach test data
detach(kola.c)
```

data(kola.c)

## Test Data for Function ltdl.fix.df

### Description

A set of test data to demonstrate how negative values are changed to half their positive value. Optionally numeric coded values representing missing data and/or zero values may be replaced by NAs.

The `.csv` file was read without deleting ID, the row (observation) identifier in the first column, from the header record. Therefore the character row ID is saved as a factor variable. If ID had been deleted from the header record the row ID would have been stored as `dimnames(fix.test)[[1]]`.

### Usage

`fix.test`

### Format

A data frame containing 15 rows and 5 columns (3 factors, one is ID, and 2 numeric).

### See Also

`fix.test.asis`

---

data(kola.c)

## Test Data for Function ltdl.fix.df

### Description

A set of test data to demonstrate how negative values are changed to half their positive value. Optionally numeric coded values representing missing data and/or zero values may be replaced by NAs.

The `.csv` file was read without deleting ID, the row (observation) identifier in the first column, from the header record, and with `as.is` set to `as.is = c(1)`. Therefore the character row ID is saved as a character variable, not as a factor variable. If ID had been deleted from the header record the row ID would have been stored as `dimnames(fix.test)[[1]]`. 
Usage

fix.test

Format

A data frame containing 15 rows and 5 columns (1 character, 2 factors, and 2 numeric).

See Also

fix.test

---

framework.stats

Compile Framework/Subset Summary Statistics

Description

Function to compile summary statistics for use with function framework.summary from the `output` of function gx.stats.

Usage

framework.stats(xx)

Arguments

xx name of the variable to be processed.

Details

The function compiles summary statistics consisting of the count of valid data, the number of NAs, the minimum, 2nd, 5th, 10th, 25th (Q1), 50th (median), 75th (Q3), 90th, 95th and 98th percentiles and the maximum. The 95% confidence interval for the median is computed via the binomial theorem. In addition the Median Absolute Deviation (MAD) and Inter-Quartile Standard Deviation (IQSD) are computed as robust estimates of the standard deviation. Finally, the mean, standard deviation and coefficient of variation as a percentage are computed.

Value

table a 20-element table is returned, see below:
[1] the data/subset (sample) size, N.
[2] number of NAs encountered in the input vector, NNA.
[3:13] the data minimum, 2nd, 5th, 10th, 25th (Q1), 50th (median), 75th (Q3), 90th, 95th and 98th percentiles and the maximum.
[14:15] the lower and upper 95% confidence bounds for the median.
[16] the Median Absolute Deviation (MAD).
[17] the Inter-Quartile Standard Deviation (IQSD).
framework.summary

[18] the data (sample) Mean.
[19] the data (sample) Standard Deviation (SD).
[20] the Coefficient of Variation as a percentage (CV%).

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df. Any NAs in the data vector are counted and then removed prior to computing the summary statistics.

Author(s)

Robert G. Garrett

See Also

gx.stats, ltdl.fix.df, remove.na

Examples

## Make test data available
data(kola.c)attach(kola.c)

## Computes summary statistics for the Cu data
fs <- framework.stats(Cu)
fs

## Computes summary statistics for Finnish subset of the Cu data
fs <- framework.stats(Cu[COUNTRY == "FIN"])
fs

## Clean-up and detach test data
rm(fs)detach(kola.c)

framework.summary Generate and Save Framework/Subset Summary Statistics

Description

Function to generate ‘framework’ or subset summary statistics and save them as a ‘.csv’ file in the R working directory. The file can be directly imported into a spreadsheet, e.g., MS Excel, for inspection, or into other software, e.g., a Geographical Information System (GIS) where the spatial information concerning the ‘framework’ units is available, e.g., ecoclassification units.

Usage

framework.summary(group, x, file = NULL)
Arguments

- **group**: the name of the factor variable by which the data are to be subset.
- **x**: name of the variable to be processed.
- **file**: the first part of the file name identifying the data source for saving the function output in the R working directory, see Details below.

Details

A default file name is generated by concatenating the data frame, group and variable, x, names, separated by underscores and .csv. If file contains text it is used as the first part of the file name identifying the data source for the file to be saved in the specified folder, for example, file = "D:\R_work\Project3\C_soils". If no folder is specified the file is saved in the R working directory.

Output to the current device is suppressed. The output file can be inspected with spreadsheet software or a viewer of the user’s choice.

Note

To set the R working directory, if it has not already been set in a first function, use at the R command line, for example:

```r
setwd("C:\\R\\Wn\n")
```

where ‘n’ is some number, which will result in all saved output being placed in that folder. The folder needs to be created before the R session commences.

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df.

Any NAs in the data vector are counted and then removed prior to computing the summary statistics.

The function framework.stats is employed to compute the summary statistics.

Author(s)

Robert G. Garrett

See Also

framework.stats, ltdl.fix.df, remove.na

Examples

```r
## Make test data available
data(kola.c)
attach(kola.c)

## Saves the file kola_c_COUNTRY_Cu_summary.csv for later use
## in the R working directory.
framework.summary(COUNTRY, Cu, file = "Kola_C_horizon")

## Detach test data
detach(kola.c)
```
**gx.2dproj**

Function to Compute and Display 2-d Projections for Data Matrices

**Description**

Function computes and displays 2-d projections of data matrices using either Sammon Non-linear Mapping (default), Multidimensional Scaling, Kruskal’s non-metric Multidimensional Scaling (see Venables and Ripley (2001) and Cox and Cox (2001)). The original S-Plus implementation also computed the Minimum Spanning Tree plane projection (Friedman and Rafsky, 1981) as it was available in the Venables and Ripley MASS library for S-Plus. However, the R implementation of the MASS library does not include Minimum Spanning Trees. In the R implementation, Projection Pursuit has been added using the fastICA procedure of Hyvarinen and Oja (2000). Provision is made to optionally trim individuals (rows) from the input data matrix.

**Usage**

```r
gx.2dproj(xx, proc = "sam", ifilr = FALSE, log = FALSE, rsnd = FALSE, snd = FALSE, range = FALSE, main = "", setseed = FALSE, row.omits = NULL, ...)
```

**Arguments**

- `xx` then by p matrix for which the 2-d projection is required.
- `proc` the 2-d projection procedure required, the default is `proc = "sam"` for Sammon Non-Linear Mapping. For Classic (metric) Multidimensional Scaling use `proc = "mds"`, for Kruskal’s non-metric Multidimensional Scaling use "iso", and for Projection Pursuit use "ica".
- `ifilr` optional isometric log-ratio transformation, the default is no transformation. Recommended for closed compositionl, geochemical, data, when `ifilr = TRUE` all other transformations are ignored.
- `log` optional (natural) log transformation of the data, the default is no log transformation. For a log transformation set `log = TRUE`.
- `rsnd` optional robust normalization of the data with matrix column medians and MADs, the default is no transformation. For a robust normalization set `rsnd = TRUE`.
- `snd` optional normalization of the data with matrix column means and standard deviations, the default is no transformation. For a normalization set `snd = TRUE`. If `rsnd = TRUE`, then `snd` will be set to `FALSE`.
- `range` optional range transformation for the matrix columns, the data values being scaled to between zero and one for, respectively, the minimum and maximum column values. If the data are range transformed, other normalization transformation requests will be ignored.
- `main` an alternative plot title, see Details below.
- `row.omits` permits rows, individuals, to be trimmed from the input matrix, the default `row.omits = NULL` is for no trimming. To trim individuals enter their row numbers as a concatenated string, e.g. `row.omits = c(13,15,16)`. The list
may be extended by adding additional row numbers so as to display the 2-d structure of the remaining core data and whether further multivariate outliers are present.

setseed sets the random number seed for fastICA so that all runs result in the same projection, and that projection is generally similar to the Sammon projection on the ilr transformed Howarth - Sinding-Larsen data set.

... further arguments to be passed to methods concerning the generated plots. For example, if smaller plotting characters are required, specify cex = 0.8; or if some colour other than black is required for the plotting characters, specify col = 2 to obtain red (see display.lty for the default colour palette). If it is required to make the plot title smaller, add cex.main = 0.9 to reduce the font size by 10%.

Details

If main is undefined a default plot title is generated by appending the input matrix name to the text string "2-d Projection for":. If no plot title is required set main = "", or if a user defined plot title is required it should be defined in main, e.g., main = "Plot Title Text".

Firstly, it is strongly recommended that if the input data matrix is for data from a closed compositional, geochemical, data matrix that an ilr transform be applied to the data, if ilr = TRUE. This has the effect of reducing the dimension of the data matrix from p to (p-1). Otherwise, it is desirable to normalize, centre and scale, or undertake a range transformation on the data to ensure the variables have equal ‘weight’ in the projections. If no transformation is requested a warning message is displayed.

The x- and y-axis labels are set appropriately to indicated the type of 2-d projection in the display. A measure of the ‘stress’ in generating the 2-d projection is estimated and displayed, low stress indicates the projection faithfully represents the relative ‘positions’ of the data in the original p-space.

Value

The following are returned as an object to be saved for further use:

main the plot title.
input a text string containing the name of the n by p matrix containing the data, and a list of the row numbers of any individuals trimmed, if none are trimmed the entry is NULL.
usage The projection option selected, and the values, TRUE or FALSE, for the ilr, log, robust normalization, normalization, and range transformation options.
xlab the 2-d projection x-axis label.
ylab the 2-d projection y-axis label.
matnames the individual, sample, row identifiers and the names of the input variables. If there are no individual, sample, row identifiers then row numbers are used. If an ilr transform has been used the variable names will be the (p-1) synthetic ilr variable names. If a trim has been executed only the row identifiers for the remaining data are stored.
row.numbers  the row numbers of the individuals, samples, remaining after a trim. If a trim has been executed only the row numbers for the remaining data are stored.

x  the n x-axis values for the 2-d projection.

y  the n y-axis values for the 2-d projection.

stress  the estimated stress of fitting 2-d projection to the p-space data.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df.

Any rows in the data matrix with with NAs are removed prior to computing the 2-d projection. In the instance of an ilr transformation NAs have to be removed prior to undertaking the transformation, see remove.na.

The results of repeated executions of the ‘fastICA’ implementation of Projection Pursuit lead to various mirror images of one another unless set.seed is used to ensure each execution commences with the same seed.

This function requires that packages MASS (Venables and Ripley) and fastICA (Marchini, Heaton and Ripley) both be available.

Author(s)
Robert G. Garrett

References


See Also

ltdl.fix.df, remove.na, gx.2dproj.plot, sammon, cmdscale, isoMDS, fastICA, set.seed

Examples

```r
## Make test data available
data(sind.mat2open)

## Display default, Sammon non-linear map, 2-d projection
sind.2dproj <- gx.2dproj(sind.mat2open, ifilr = TRUE)
```
## gx.2dproj.plot

### Function to Display a Saved 2-d Projection Object

**Description**

Displays the 2-d projection saved from `gx.2dproj`, optionally the row numbers of the input matrix or the row identifiers, if available, may be displayed instead of the default plotting symbol.

**Usage**

```r
gx.2dproj.plot(save, rowids = NULL, main = "", pch = 3, cex = 0.8, col = 1, ...)
```

**Arguments**

- `save` the saved object from `gx.2dproj`.
- `rowids` to display the input matrix row numbers set `rowids = TRUE`. Setting `rowids = FALSE` replaces the row numbers with the row identifiers from the input matrix. The default `rowids = NULL` causes the default symbol, a ‘+’ to be plotted.
- `main` an alternative plot title to that in the saved object from `gx.2dproj`, see Details below.
- `pch` by default a ‘+’, `pch = 3` will be plotted. See `Note` below for alternate plotting symbols.
- `cex` by default symbols and characters are plotted at `cex = 0.8`, a 20% reduction in font size.
- `col` by default symbols and characters are plotted in black, `col = 1`, specify `col = 2` to obtain red (see `display.lty` for the default colour palette). See `Note` below for alternate colours.

---

```r
## Display saved object identifying input matrix row numbers (cex = 0.7),
## and with an alternate main title (cex.main = 0.8)
gx.2dproj.plot(sind.2dproj, rowids = TRUE, cex = 0.7, cex.main = 0.8,
main = "Howarth & Sinding-Larsen\nStream Sediment ilr Transformed Data")

## Display Kruskal's non-metric multidimensional scaling 2-d projection
sind.2dproj <- gx.2dproj(sind.mat2open, proc = "iso", ifillr = TRUE)

## Display saved object identifying input matrix row numbers (cex = 0.7),
## and with an alternate main title (cex.main = 0.8)
gx.2dproj.plot(sind.2dproj, rowids = FALSE, cex = 0.7, cex.main = 0.8,
main = "Howarth & Sinding-Larsen\nStream Sediment ilr Transformed Data")

## Display default, Sammon non-linear map, 2-d projection, removing the three
## most extreme individuals
sind.2dproj.trim3 <- gx.2dproj(sind.mat2open, ifillr = TRUE, row.omits = c(13,15,16))

## Clean-up
rm(sind.2dproj)
rm(sind.2dproj.trim3)
```
... further arguments to be passed to methods concerning the plot. For example, if it is required to make the plot title smaller, add `cex.main = 0.9` to reduce the font size by 10%.

Details

If `main` is undefined the plot title from the saved object from `gx.2dproj` is displayed. If no plot title is required set `main = ""`, or if a user defined plot title is required it should be defined in `main`, e.g., `main = "Plot Title Text"`.

The x- and y-axis labels are those in the saved object from `gx.2dproj` and indicate the type of 2-d projection in the display.

Note

The available symbols are:
- `pch`: 0 = square, 1 = circle, 2 = triangle, 3 = plus, 4 = X, 5 = diamond, 6 = upside-down triangle, 7 = square with X, 8 = asterisk, 9 = diamond with plus, 10 = circle with plus, 11 = double triangles, 12 = square with plus, 13 = circle with X, 14 = square with upside-down triangle.
- Symbols 15 to 18 are solid in the colour specified:
- 15 = square, 16 = circle, 17 = triangle, 18 = diamond.

The available colours from the default ‘palette’ are:
- `Col`: 0 = none, 1 = black, 2 = red, 3 = green, 4 = dark blue, 5 = turquoise, 6 = pink, 7 = yellow, 8 = grey, 9 = black.

Author(s)

Robert G. Garrett

References


See Also

gx.2dproj

Examples

```r
## Make test data available
data(sind.mat2open)

## Display default 2-d projection
sind.save <- gx.2dproj(sind.mat2open, if1lr = TRUE)
```
## Description

This is an internal function used to plot fences at stated probability levels on a Chi-square plot to assist in the assessment of the plotted distribution. By default fences are plotted for the 90th, 95th and 98th percentiles of the Chi-square distribution. The function is called from `gx.md.plt()`, itself called from `gx.md.plot` that is used to display Chi-square plots generated by `gx.mva`, `gx.mva.closed`, `gx.robmva`, `gx.robmva.closed`, `gx.md.gait` and `gx.md.gait.closed`.

## Usage

```
gx.add.chisq(p = c(0.98, 0.95, 0.9), df = NULL, ifflip = FALSE, 
cex = 0.6)
```

## Arguments

- **p**
  - the percentiles of the Chi-square distribution for the fences to be displayed, by default the 90th, 95th and 98th percentiles. If no fences are required set `p = NULL`.
- **df**
  - the degrees of freedom for the Chi-square distribution, the number of variables in the multivariate distribution.
- **ifflip**
  - by default fence labelling is placed to the left of the fences just above the x-axis. Setting `ifflip = TRUE` places the annotation to the right.
- **cex**
  - the scale expansion factor for the fence labelling, by default `cex = 0.6`.

## Author(s)

Robert G. Garrett
See Also

gx.md.plot, gx.md.gait, gx.md.gait.closed

Examples

```r
## Synthesize test data
test <- mvrnorm(100, mu = c(40, 30), Sigma = matrix(c(6, 3, 2), 2, 2))

## Display annotated Chi-square plot
gx.md.gait(test)
gx.md.gait(test, ifadd = c(0.9, 0.98))

## Clean-up
rmm(test)
```

gx.adjr2  

Function to compute Adjusted r-squared values

Description

Function to compute the Adjusted R-square value from the Multiple R-squared value displayed in the summary of a lm object. See Note below.

Usage

gx.adjr2(mr2, n, p)

Arguments

- `mr2`: the Multiple R-squared value.
- `n`: the number of cases in the regression model.
- `p`: the number of independent (explanatory or predictor) variables in the model.

Note

The Adjusted R-squared value is a long established criterion. It may be calculated casually by this function, or may be extracted from a lm object, using summary(lm.object[[9]]). However, users are urged to investigate Akaike’s Information Criterion, AIC, as a procedure for comparing the fits of alternate models, and the use of the step function for automated model selection.

Author(s)

Robert G. Garrett

See Also

summary, AIC, step
Examples

gx.adjr2(0.7394, 111, 11)
gx.adjr2(0.713, 111, 6)

Description

Displays cumulative normal probability (CPP) plots for up to nine data subsets, using combinations of symbols and colours to identify each subset. Note CPP plots are equivalent to Q-Q plots and are more frequently used by physical scientists and engineers.

Usage

gx.cnpplts(xlab = "", log = FALSE, xlim = NULL, main = "", iflgnd = FALSE, ...)

Arguments

xlab a title for the x-axis must be provided, even if it is ‘no title’, i.e. xlab = "", or an informative title may be provided, see Examples.

log log must be specified, TRUE or FALSE. If it is required to display the data with logarithmic (x-axis) scaling, set log = TRUE. If log scaling is not required, set log = FALSE.

xlim if the internally generated values for xlim are to be replaced see the Note below. If the internally generated x-axis limits are satisfactory omit any reference to xlim in the call to the function.

main a title must be provided, even if it is ‘no title’, i.e. main = "". If main is specified a title will be added above the plot, e.g., main = "Kola Project, 1995".

iflgnd iflgnd must be specified, TRUE or FALSE. If a R generated legend is to be be placed on the plot, set iflgnd = TRUE. On completion of CPP plotting the cursor is activated, locate it at the top left of the space where the legend is to be added and ‘left button’ on the pointing device. The legend comprises the symbol/colour combination, the name of the subset plotted and the data subset size; this information is also displayed on the current device. If no legend is required, set iflgnd = FALSE.

... the names of the data subsets (objects), separated by commas, to be plotted, up to a maximum of nine. See the example below for subset pre-processing steps that lead to a more presentable legend.
Details

Unlike most other functions in ‘rgr’ all the arguments must be specified explicitly, except xlim. This is the cost of being able to append up to nine subset names in the function call. The function needs to know where subset names start in the list passed to the function.

A default allocation of symbols and colours, and the size of the legend text, is provided in gx.cnpplts.setup. These may be edited if required, they are imported into gx.cnpplts at function run time.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df. Any NAs in the data vector are removed prior to displaying the plot.

Although this function is most frequently used to compare the frequency distributions for the same element in multiple subsets of the data, it may also be used to display frequency distributions for multiple elements.

If it is required to set the x-axis limits to specific values they can be set, e.g., xlim = c(0, 200) or c(2, 200), the latter being appropriate for a logarithmically scaled plot, i.e. log = TRUE. If the defined limits lie within the observed data range a truncated plot will be displayed. Setting the limits wider than the default limits can provide additional space for annotation of the display.

By setting iflgn = FALSE no internally generated legend will be added. Alternately, a legend can be constructed with the text function and placed with the locator at execution of the text function.

Author(s)

Robert G. Garrett

See Also

gx.cnpplts.setup, display.marks, display.lty, ltdl.fix.df, text

Examples

```r
## Make test data available
data(kola.c)
attach(kola.c)

## An example
gx.cnpplts(xlab = "Cu (mg/kg) in <2 mm Kola C-horizon soil", log = TRUE,
xlim = NULL, main = "", iflgn = FALSE, Cu[COUNTRY == "RUS"],
Cu[COUNTRY == "FIN"], Cu[COUNTRY == "NOR"])

## An example that leads to a cleaner legend
## First select data for the variable to be plotted for the subsets, from
dimnames(kola.c) we know that Be is the 19th column in the data frame
Norway <- gx.subset(kola.c,COUNTRY=="NOR")[,19]
Russia <- gx.subset(kola.c,COUNTRY=="RUS")[,19]
```
Finland <- gx.subset(kola.c,COUNTRY=="FIN")[,19]
gx.cnpplt(xlab = "Be (mg/kg) in <2 mm Kola C-horizon soils", log = TRUE, 
xlim = NULL, main = "", iflgnd = FALSE, Finland, Russia, Norway)

## An example where the limits of the x-axis are provided

gx.cnpplt(xlab = "Be (mg/kg) in <2 mm Kola C-horizon soils", log = TRUE, 
xlim = c(0.02, 20), main = "", iflgnd = FALSE, Finland, Russia, Norway)

## An example of a multi-element display

gx.cnpplt(xlab = "Concentrations (mg/kg) in <2 mm Kola C-horizon soils", 
log = TRUE, xlim = NULL, main = "Kola Project, 1995", 
iflgnd = FALSE, Cu, Ni, Co)

## Clean-up and detach test data
rm(Norway)
rm(Russia)
rm(Finland)
detach(kola.c)

---

### gx.cnpplt

**Set Up and Display Symbolsy for function gx.cnpplt**

**Description**

Permits a user to display the symbol mark and colour combinations to be used in function `gx.cnpplt`, and change them and the legend font size, if required. Any changes require editing the function and some elementary R-scripting skills, see Note below.

**Usage**

`gx.cnpplt.setup(display = FALSE)`

**Arguments**

- `display` if `display = TRUE` the symbol mark and colour combinations are displayed on the current device. If `display = FALSE` output is suppressed.

**Details**

The available symbols are:

- `pch`: 0 = square, 1 = circle, 2 = triangle, 3 = plus, 4 = X,
- 5 = diamond, 6 = upside-down triangle, 7 = square with X,
- 8 = asterisk, 9 = diamond with plus, 10 = circle with plus,
- 11 = double triangles, 12 = square with plus,
- 13 = circle with X, 14 = square with upside-down triangle.

Symbols 15 to 18 are solid in the colour specified:

- 15 = square, 16 = circle, 17 = triangle, 18 = diamond.

The available colours from the default 'palette' are:

- `Col`: 0 = none, 1 = black, 2 = red, 3 = green, 4 = dark blue,
- 5 = turquoise, 6 = pink, 7 = yellow, 8 = grey, 9 = black.
Value

- **pchs**: a vector of 9 elements defining the symbols, marks, to use for plotting the 1 to 9 permissible subsets.
- **symcols**: a vector of 9 elements defining the colours from the ‘default’ palette to use for the colours of the 1 to 9 permissible subset symbols.
- **cex**: the text scale expansion factor to use in the optional legend for function `gx.cnplts`, the default is 0.8.
- **cexp**: the scale expansion factor for the plotting symbols in function `gx.cnplts`, the default is 0.9.

Note

To edit the function use `fix(gx.cnplts.setup)` to extract a copy of the function from the ‘rgr’ library for editing. It will help to have a colour printed copy of the display, `display = TRUE`, from this function at hand. Note that after editing and saving the function will remain in the workspace and you may get warning messages that can be ignored.

To return to the defaults delete the edited function from the workspace, i.e. `rm(gx.cnplts.setup)`.

Author(s)

Robert G. Garrett

See Also

`display.marks`, `points`, `display.lty`

---

**gx.ecdf**

*Empirical Cumulative Distribution Function (ECDF)*

Description

Displays an empirical cumulative distribution function (ECDF) plot with a zero-to-one linear y-scale as part of the multi-panel display provided by `shape`. The function may also be used stand-alone.

Usage

```r
gx.ecdf(xx, xlab = deparse(substitute(xx)), ylab = "Empirical Cumulative Distribution Function", log = FALSE, xlim = NULL, main = "", pch = 3, ifqs = FALSE, cex = 0.8, ...)
```
Arguments

- **xx**: name of the variable to be plotted.
- **xlab**: by default the character string for xx is used for the x-axis title. An alternate title can be displayed with xlab = "text string", see Examples.
- **ylab**: a title for the y-axis, defaults to "Empirical Cumulative Distribution Function".
- **log**: if it is required to display the data with logarithmic (x-axis) scaling, set log = TRUE.
- **xlim**: when used in the shape function, xlim is determined by gx.hist and used to ensure all four panels in shape have the same x-axis scaling. However, when used stand-alone the limits may be user-defined by setting xlim, see Note below.
- **main**: when used stand-alone a title may be added optionally above the plot by setting main, e.g., main = "Kola Project, 1995".
- **pch**: by default the plotting symbol is set to a plus, pch = 3, alternate plotting symbols may be chosen from those displayed by display.marks, see also Notes below.
- **ifqs**: setting ifqs = TRUE results in horizontal and vertical dotted lines being plotted at the three central quartiles and their values, respectively.
- **cex**: by default the size of the text for data set size, N, is set to 80%, i.e. cex = 0.8, and may be changed if required.
- **...**: further arguments to be passed to methods. The colour of the plotting symbols may be changed from default black, e.g., col = 2 for red. The size of the axis scale annotation can be change by setting cex.axis, the size of the axis titles by setting cex.lab, and the size of the plot title by setting cex.main. For example, if it is required to make the plot title smaller, add cex.main = 0.9 to reduce the font size by 10%.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df. Any NAs in the data vector are removed prior to displaying the plot.

Although the cumulative normal percentage probability (CPP) plot is often the preferred method for displaying the cumulative data distribution as it provides greater detail for inspection in the tails of the data, the ECDF is particularly useful for studying the central parts of data distributions as it has not been compressed to make room for the scale expansion in the tails of a cumulative normal percentage probability (CPP) plot.

If the default selection for xlim is inappropriate it can be set, e.g., xlim = c(0, 200) or c(2, 200), the latter being appropriate for a logarithmically scale plot, i.e. log = TRUE. If the defined limits lie within the observed data range a truncated plot will be displayed. If this occurs the number of data points omitted is displayed below the total number of observations.

The available symbols are:

- **pch**: 0 = square, 1 = circle, 2 = triangle, 3 = plus, 4 = X, 5 = diamond, 6 = upside-down triangle, 7 = square with X, 8 = asterisk, 9 = diamond with plus, 10 = circle with plus, 11 = double triangles, 12 = square with plus,
13 = circle with X, 14 = square with upside-down triangle.
Symbols 15 to 18 are solid in the colour specified:
15 = square, 16 = circle, 17 = triangle, 18 = diamond.

If it is desired to prepare a display of data falling within a defined part of the actual data range, then
either a data subset can be prepared externally using the appropriate R syntax, or xx may be defined
in the function call as, for example, Cu[Cu < some.value] which would remove the influence of
one or more outliers having values greater than some.value. In this case the number of data values
displayed will be the number that are <some.value.

Author(s)
Robert G. Garrett

See Also
display.marks, ltdl.fix.df, remove.na

Examples

```r
## Make test data available
data(kola.o)
attach(kola.o)

## Plot a simple ECDF
gx.ecdf(Cu)

## Plot an ECDF with more appropriate labelling and with the quartiles
## indicated
gx.ecdf(Cu, xlab = "Cu (mg/kg) in <2 mm Kola 0-horizon soil", log = TRUE,
ifqs = TRUE)

## Detach test data
detach(kola.o)
```

---

**gx.fRACTile**

*Estimate the Fractile for a Specified Quantile*

**Description**

Estimates the fractile for a specified quantile of a data set by linear interpolation from the ranked
data. If the function is run as temp <- gx.fRACTile(xx, q) the fractile is not displayed, but
retained in temp for subsequent use or display.

**Usage**

```r
gx.fRACTile(xx, q, display = TRUE)
```
Arguments

xx the data set for which the quantile is to be estimated.
q the fractile (value) for which the quantile is required, must be within the range of xx.
display the default is to display the quantile and estimated fractile on the current device. If no display is required, set display = FALSE.

Value

f the estimated fractile.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df.

Any NAs in the data vector are removed prior to preparing the boxplots.

Author(s)

Based on a script shared on S-News by Nick Ellis, April 2002

See Also

ltdl.fix.df, remove.na, gx.quantile

Examples

## Make test data available
data(kola.o)
attach(kola.o)

## Estimate the fractile for 20 mg/kg As
gx.fractile(As, 20)
temp <- gx.fractile(As, 20)
temp

## Clean-up and detach test data
rm(temp)
detach(kola.o)
gx.hist  Plot a Histogram

Description

Plots a histogram for a data set, the user has options for defining the axis and main titles, the x-axis limits, arithmetic or logarithmic x-axis scaling, the number of bins the data are displayed in, and the colour of the infill.

Usage

gx.hist(xx, xlab = deparse(substitute(xx)),
ylab = "Number of Observations", log = FALSE, xlim = NULL,
main = "", nclass = "Scott", colr = NULL, ifnright = TRUE,
cex = 1, ...)

Arguments

xx  name of the variable to be plotted
xlab  by default the character string for xx is used for the x-axis title. An alternate title can be displayed with xlab = "text string", see Examples.
ylab  a default y-axis title of "Number of Observations" is provided, this may be changed, e.g., ylab = "Counts".
log  if it is required to display the data with logarithmic (x-axis) scaling, set log = TRUE.
xlim  default limits of the x-axis are determined in the function for use in other panel plots of function shape. However, when used stand-alone the limits may be user-defined by setting xlim, see Note below.
main  when used stand-alone a title may be added optionally above the plot by setting main, e.g., main = "Kola Project, 1995".
nclass  the default procedure for preparing the histogram is to use the Scott (1979) rule. This usually provides an informative histogram, other optional rules are nclass = "Sturges" or nclass = "FD"; the later standing for Freedman-Diaconis (1981), a rule that is resistant to the presence of outliers in the data. See Venables and Ripley (2001) for details.
colr  by default the histogram is infilled in grey, colr = 8. If no infill is required, set colr = 0. See function display.lty for the range of available colours.
ifnright  controls where the sample size is plotted in the histogram display, by default this in the upper right corner of the plot. If the data distribution is such that the upper left corner would be preferable, set ifnright = FALSE.
cex  by default the size of the text for data set size, N, is set to 80%, i.e. cex = 0.8, and may be changed if required.
...  further arguments to be passed to methods. For example, the size of the axis titles may be changed by setting cex.lab, the size of the axis labels by setting cex.axis, and the size of the plot title by setting cex.main. For example, if it is required to make the plot title smaller, add cex.main = 0.9 to reduce the font size by 10%.
Value

xlim A two element vector containing the actual minimum [1] and maximum [2] x-axis limits used in the histogram display are returned. These are use in function shape to ensure all panels have the same x-axis limits.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df. Any NAs in the data vector are removed prior to displaying the plots.

If the default selection for xlim is inappropriate it can be set, e.g., xlim = c(0, 200) or c(2, 200), the latter being appropriate for a logarithmically scaled plot, i.e. log = TRUE. If the defined limits lie within the observed data range a truncated plot will be displayed. If this occurs the number of data points omitted is displayed below the total number of observations.

If it is desired to prepare a display of data falling within a defined part of the actual data range, then either a data subset can be prepared externally using the appropriate R syntax, or xx may be defined in the function call as, for example, Cu[Cu < some.value] which would remove the influence of one or more outliers having values greater than some.value. In this case the number of data values displayed will be the number that are <some.value.

Author(s)

Robert G. Garrett

References


See Also

display.lty, ltdl.fix.df, remove.na

Examples

## Make test data available
data(kola.o)
attach(kola.o)

## Generates an initial display to have a first look at the data and
## decide how best to proceed
gx.hist(Cu)

## Provides a more appropriate initial display
gx.hist(Cu, xlab = "Cu (mg/kg) in <2 mm Kola O-horizon soil", log = TRUE)

## Causes the Friedman-Diaconis rule to be used to select the number
## of histogram bins

gx.hist(Cu, xlab = "Cu (mg/kg) in <2 mm Kola O-horizon soil", log = TRUE,
The hypergeometric distribution is used to infer if the number of anomalous sites along a traverse reliably reflect the presence of the dispersion pattern from a known mineral occurrence. The function displays the probability of the observed outcome could be due to chance alone.

Usage

gx.hypergeom(tt, aa, kk, xx)

Arguments

* tt total number of sites along a traverse.
* aa number of sites that a priori should be anomalous.
* kk total number of > threshold sites.
* xx number of the aa that are > threshold.

Details

See Stanley (2003) for details, the examples below reproduce the results in Table 1 and Table 2.

Note

Effectively, the hypothesis being tested is that the pattern of above threshold (see fences), sites coincides the the expected dispersion pattern from a known mineral occurrence. This requires that the geochemist uses knowledge of the dispersion processes active along the traverse, both chemical and mechanical, to predict an expected dispersion pattern.

Author(s)

Robert G. Garrett

References


See Also

gx.runs
Examples

## From Stanley (2003) Tables 1 and 2

```r
gx.hypergeom(31, 10, 5, 3)
gx.hypergeom(31, 10, 3, 2)
gx.hypergeom(31, 10, 4, 3)

# Example with hypergeom

gx.hypergeom(31, 10, 4, 4)
gx.hypergeom(31, 10, 6, 5)
gx.hypergeom(31, 10, 3, 3)
```

---

**gx.ks.test**

*Kolmogorov-Smirnov test with ECDF Plot*

---

**Description**

Function to plot the Empirical Cumulative Distribution Functions (ECDFs) of two distributions and undertake a Kolmogorov-Smirnov test for the Hypothesis that both distributions were drawn from the same underlying distribution.

**Usage**

```r
gx.ks.test(xx1, xx2, xlab = "", x1lab = deparse(substitute(xx1)),
x2lab = deparse(substitute(xx2)),
ylab = "Empirical Cumulative Distribution Function", log = FALSE,
main = "", pch1 = 3, col1 = 2, pch2 = 4, col2 = 4,
ifresult = TRUE, cex = 0.8, cexp = 0.9, ...)
```

**Arguments**

- `xx1` name of the first variable to be plotted - distribution to be tested.
- `xx2` name of the second variable to be plotted - distribution to be tested.
- `xlab` a title for the x-axis, by default none is provided. For example, `xlab = "Cu (mg/kg) in <2 mm C-horizon soil"`.
- `x1lab` the name for the first distribution to be plotted, defaults to `x1lab = deparse(substitute(xx1))`.
- `x2lab` the name for the second distribution to be plotted, defaults to `x2lab = deparse(substitute(xx2))`.
- `ylab` defaults to `ylab = "Empirical Cumulative Distribution Function"` and may be changed if required.
- `log` if it is required to display the data with logarithmic (x-axis) scaling, set `log = TRUE`. The Kolmogorov-Smirnov test is undertaken on untransformed data. If it is to be undertaken on transformed data, the transformation should be applied previously or in the call, e.g., `log10(xx1), sqrt(xx1), etc.`
main
   a plot title if one is required, e.g., main = "Kola Ecogeochmstry Project, 1995".

pch1
   the plotting symbol for the first distribution, defaults to a '+' sign, pch = 3, and
   may be changed if required, see display.marks.

col1
   the colour of the plotting symbol for the first distribution, defaults to red, col1 = 2,
   and may be changed if required, see display.lty.

pch2
   the plotting symbol for the second distribution, defaults to a 'x' sign, pch = 4,
   and may be changed if required, see display.marks.

col2
   the colour of the plotting symbol for the second distribution, defaults to blue, 
col2 = 4, and may be changed if required, see display.lty.

ifresult
   setting ifresult = FALSE suppresses the ability to add the results of the
   Kolmogorov-Smirnov test to the plot, the default is ifresult = TRUE.

cex
   the scaling factor for the test results and legend identifying the symbology for 
   each distribution and its population size is set to cex = 0.8 by default, it may 
   be changed if required.

cexp
   the scaling factor for the plotting symbol size is set to cexp = 0.9 by default, 
   if may be changed if required.

... further arguments to be passed to methods. For example, the size of the axis 
   scale annotation can be change by setting cex.axis, the size of the axis titles 
   by seeting cex.lab, and the size of the plot title by setting cex.main. For 
   example, if it is required to make the plot title smaller, add cex.main = 0.9 to 
   reduce the font size by 10%.

Details

By default the results of the Kolmogorov-Smirnov test are added to the plot. On completion of
the ECDF plotting the cursor is activated, locate it at the centre of the area where the results are to
added and ‘left button’ on the pointing device. When ifresult = FALSE the cursor is not activated
for this annotation; this is sometimes convenient if there is insufficient space for the results without
overprinting on the ECDFs and report quality plots are required. Also by default a legend is added
to the plot, the cursor is activated and should be placed at the top left corner of area where the legend
is to be added and ‘left button’ on the pointing device. The legend consists of two lines indicating
the symbology (symbol and colour), name and size of each distribution.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes
representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df.
Any NAs in the data vectors are removed prior to displaying the plot and undertaking the Kolmogorov-
Smirnov test.

Author(s)

Robert G. Garrett

See Also

gx.cnppltls, display.marks, display.lty, ltdl.fix.df, text
Examples

```r
## Make test data available
data(kola.c)
attach(kola.c)

## First select data for the variable to be plotted for the subsets, from
dimnames(kola.c) we know that Be is the 19th column in the data frame
Norway <- gx.subset(kola.c,COUNTRY=="NOR")[,19]
Russia <- gx.subset(kola.c,COUNTRY=="RUS")[,19]
Finland <- gx.subset(kola.c,COUNTRY=="FIN")[,19]

## NOTE: the examples below are commented out as gx.ks.test makes a
call to the locator function that fails when the examples are run
## during package checking and building
## Initial plot
## gx.ks.test(Finland, Russia, xlab = "Be (mg/kg) in <2 mm Kola C-horizon soils",
## log = TRUE, main = "Kola Ecogeochemistry Project, 1995")

## The same plot as above, but with the results suppressed and the
## annotation better scaled, the legend and plot symbols at 75%, the
## plot title at 90% and the axis labelling at 80%
## gx.ks.test(Finland, Russia, xlab = "Be (mg/kg) in <2 mm Kola C-horizon soils",
## log = TRUE, main = "Kola Ecogeochemistry Project, 1995",
## ifresult = F, cex = 0.75, cexp = 0.75, cex.main = 0.9, cex.lab = 0.8,
## cex.axis = 0.8)

## Clean-up and detach test data
rm(Norway)
rm(Russia)
rm(Finland)
detach(kola.c)
```

gx.lm.vif

### Estimate Variance Inflation Factor (VIFs)

**Description**

Function estimates Variance Inflation Factors (VIFs), measures of collinearity in a linear model. The VIF provides a measure of how much the variance of an estimated regression coefficient is increased because of collinearity. Collinearity is present when there is a high correlation between the independent, predictor, variables in a model, i.e. they tell the same ‘story’. Where collinearity exists it is often best to remove predictor variables with high VIFs from the model.

**Usage**

```r
gx.lm.vif(object, ...)```
**Arguments**

- **object**: a `lm` object.
- **...**: any additional parameters.

**Value**

A (structure) table of Variable Inflation Factors for the predictor variables.

**Note**

VIFs >5 are indicative of collinearity, and the information conveyed in that variable is also in the subset of the remaining variables.

**Author(s)**

W.N. Venables, function shared on S-News, October 21, 2002

**References**


**Examples**

```r
## Make test data available
data(sind)
attach(sind)

## Model 1
sind.1 <- lm(log(Zn) ~ Fe + log(Mn) + log(Cu) + log(Cd))
summary(sind.1)
gx.lm.vif(sind.1)

## Model 2
sind.2 <- lm(log(Zn) ~ Fe + log(Mn))
summary(sind.2)
gx.lm.vif(sind.2)
AIC(sind.1, sind.2)

## Model 3
sind.3 <- lm(log(Zn) ~ log(Mn) + log(Cu))
summary(sind.3)
gx.lm.vif(sind.3)
AIC(sind.1, sind.2, sind.3)

## Clean-up and detach test data
rm(sind.1)
rm(sind.2)
rm(sind.3)
detach(sind)
```
gx.md.display

Function to Display Membership Probabilities and Other Relevant Data

Description

Function to display the Mahalanobis distances (MDs) and predicted probabilities of membership (ppm or p_gm), together with other relevant data, following computations by functions gx.md.gait, gx.md.gait.closed, gx.mva, gx.mva.closed, gx.robmva or gx.robmva.closed. The user may select the predicted probability of membership below which the results are displayed. A simpler presentation is available with gx.md.print. Optionally the entire generated table may be saved as a `.csv` file for future use.

Usage

```r
gx.md.display(xx, pcut = 0.1, file = NULL)
```

Arguments

- **xx**
  - the data to be displayed in a `cbind` construct, see Details below.
- **pcut**
  - the probability of group membership below which records will be displayed on the current device in ascending order of membership probability, i.e. most outlying individuals first.
- **file**
  - the file name for saving the function output in the R working directory, see Details below.

Details

The data frame from which the matrices were derived for use by the above listed functions must be attached if such items as sample IDs, coordinates and data values are to be displayed. Those items and/or variables to be displayed must be appended in a `cbind` construct following the ‘MDs’ and ‘ppms’ extracted from the saved objects from the above listed functions. For example, `cbind(save.sind$md, save.sind$ppm, ID, Zn, Cu, Cd, Fe, Mn)`. The table generated by the function may be saved as a `.csv` file in the working directory, with the `.csv` being appended in the function. See example below. If `file = ""` or `file = " "` a default file name is generated as “MDs_&_variables.csv”.

Value

The displayed table, `table.rows`, is returned and may be saved as an object if required. It will contain the information passed to the function as `xx` sorted by MD and with appropriate column headings.

Note

This function is similar in purpose to `gx.mvalloc.print` for displaying multivariate outliers, however, it operates on a single population.
gx.md.gait

Author(s)

Robert G. Garrett

See Also

gx.md.gait, gx.md.gait.closed, gx.mva, gx.mva.closed, gx.robmva, gx.robmva.closed, gx.mvalloc, gx.mvalloc.print, gx.md.print.

Examples

```r
## Make test data available
data(sind.mat2open)
data(sind)
attach(sind)
## data frame sind attached to provide access to row IDs

## Estimate and display robust Mahalanobis distances
sind.save <- gx.md.gait.closed(sind.mat2open, mcdstart = TRUE, ifadd = NULL)
gx.md.display(cbind(sind.save$md, sind.save$ppm, ID, Zn, Cu, Cd, Fe, Mn), pcut = 0.3)

## Save display for future use
ngx.md.display(cbind(sind.save$md, sind.save$ppm, ID, Zn, Cu, Cd, Fe, Mn), file = "sind.save.ilr.mds")

## Clean-up
rm(sind.save)
detach(sind)
```

---

**gx.md.gait**  
*Function for Multivariate Graphical Adaptive Interactive Trimming*

**Description**

Function to undertake the GAIT (Graphical Adaptive Interactive Trimming) procedure for multivariate distributions through Chi-square plots of Mahalanobis distances (MDs) as described in Garrett (1988, 1989). For closed compositional, geochemical, data sets use `gx.md.gait.closed`. To carry out GAIT the function is called repeatedly with the weights from the previous iteration being used as a starting point. Either a percentage based MVT or a MCD robust start may be used as the first iteration.

**Usage**

```
gx.md.gait(xx, wts = NULL, trim = -1, mvtstart = FALSE, mcdstart = FALSE, main = deparse(substitute(xx)), ifadd = c(0.98, 0.95, 0.9), cexf = 0.6, cex = 0.8, ...)
```
Arguments

xx
the n by p matrix for which the Mahalanobis distances are required.
wts
the vector of weights for the n individuals, either zero or 1.
trim
the desired trim: trim < 0 - no trim, the default; trim >0 & <1 - the fraction, 0 to 1 proportion, of individuals to be trimmed; trim >= 1 - the number of individuals with the highest MDs from the previous iteration to trim.
mvtstart
set mvtstart = TRUE for a percentage based MVT (multivariate trim) start.
mcdstart
set mcdstart = TRUE for a minimum covariance determinant (mcd) robust start.
main
an alternative plot title to the default input data matrix name, see Details below.
ifadd
if probability based fences are to be displayed on the Chi-square plots enter the probabilities here, see Details below. For no fences set ifadd = NULL.
cexf
the scale expansion factor for the Ch-square fence annotation, by default cexf = 0.6.
cex
the scale expansion factor for the symbols and text annotation within the ‘frame’ of the Chi-square plot, by default cex = 0.8.
...
further arguments to be passed to methods concerning the generated plots. For example, if some colour other than black is required for the plotting characters, specify col = 2 to obtain red (see display.1ty for the default colour palette). If it is required to make the plot title or axis labelling smaller, add cex.main = 0.9 or cex.lab = 0.9, respectively, to reduce the font size by 10%.

Details

If main is undefined the name of the matrix object passed to the function is used as the plot title. This is the recommended procedure as it helps to track the progression of the GAIT. Alternate plot titles can be defined if the final saved object is passed to `gx.md.plot`. If no plot title is required set main = " ", or if a user defined plot title is required it may be defined, e.g., main = "Plot Title Text".

By default three fences are placed on the Chi-square plots at probabilities of membership of the current ‘core’ data subset, or total data if appropriate, with ifadd = c(0.98, 0.95, 0.9). Alternate probabilities may be defined as best for the display. If no fences are required set ifadd = NULL.

The Mahalanobis distance, Chi-square, plot x-axis label is set appropriately to indicated the type of robust start or trim using the value of proc.

Value

The following are returned as an object to be saved for the next iteration or final use:

main
by default (recommended) the input data matrix name.
input
the data matrix name, input = deparse(substitute(xx)), retained to be used by post-processing display functions.
matnames
the row numbers and column headings of the input matrix.
proc
the procedure followed for this iteration, used for subsequent Chi-square plot x-axis labelling.
wts the vector of weights for the \( n \) individuals, either zero or 1.

\( n \) the total number of individuals (observations, cases or samples) in the input data matrix.

\( ptrim \) the percentage, as a fraction, of samples called to be trimmed in this iteration, otherwise \( ptrim = -1 \).

\( mean \) the length \( p \) vector of means for the ‘core’ data following the current GAIT step.

\( cov \) the \( p \times p \) covariance matrix for the ‘core’ data following the current GAIT step.

\( sd \) the length \( p \) vector of standard deviations for the ‘core’ data following the current GAIT step.

\( md \) the vector of Mahalanobis distances for all the \( n \) individuals following the current GAIT step.

\( ppm \) the vector of predicted probabilities of membership for all the \( n \) individuals following the current GAIT step.

**Note**

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Any rows in the data matrix with \( \text{NAs} \) are removed prior to computations. In the instance of a log-ratio, e.g., `ilr`, transformation \( \text{NAs} \) are removed.

Warnings are generated when the number of individuals (observations, cases or samples) falls below \( 5p \), and additional warnings when the number of individuals falls below \( 3p \). At these low ratios of individuals to variables the shape of the \( p \)-space hyperellipsoid is difficult to reliably define, and therefore the results may lack stability. These limits \( 5p \) and \( 3p \) are generous, the latter especially so; many statisticians would argue that the number of individuals should not fall below \( 9p \), see Garrett (1993).

**Author(s)**

Robert G. Garrett

**References**


**See Also**

`ltdl.fix.df, remove.na, gx.md.plot, gx.md.print`
Examples

```r
## Note, the example below is presented for historical continuity. It is not recommended that this procedure be used for geochemical data. For such data function `gx.md.gait.closed` should be employed. However, the example below is consistent with Garrett (1989).
## Make test data available
data(sind)
attach(sind)
sind.mat <- as.matrix(sind[, -c(1:3)])

## Undertake original published GAIT procedure
gx.md.gait(sind.mat)
sind.gait.1 <- gx.md.gait(sind.mat, trim = 0.24, ifadd = 0.98)
sind.gait.2 <- gx.md.gait(sind.mat, wts = sind.gait.1$wts, mvtstart = TRUE, trim = 4, ifadd = 0.98)
sind.gait.3 <- gx.md.gait(sind.mat, wts = sind.gait.2$wts, trim = 1, ifadd = 0.9)
sind.gait.4 <- gx.md.gait(sind.mat, wts = sind.gait.3$wts, trim = 2, ifadd = 0.9)

## Display saved object with alternate main titles and list outliers
## IDEAS procedure
gx.md.plot(sind.gait.4, main = "Howarth & Sinding-Larsen\nStream Sediments, IDEAS procedure", cex.main = 0.8, ifadd = 0.9)
gx.md.print(sind.gait.4, pcut = 0.2)

## Clean-up and detach test data
rm(sind.mat)
rm(sind.gait.1)
rm(sind.gait.2)
rm(sind.gait.3)
rm(sind.gait.4)
detach(sind)
```

---

**gx.md.gait.closed**

*Function for Multivariate Graphical Adaptive Interactive Trimming with Compositional Data*

**Description**

Function to undertake the GAIT (Graphical Adaptive Interactive Trimming) procedure for multivariate distributions through Chi-square plots of Mahalanobis distances (MDs) as described in Garrett (1988), but for closed compositional, geochemical, data. To carry out GAIT the function is called repeatedly with the weights from the previous iteration being used as a starting point. Either a percentage based MVT or a MCD robust start may be used as the first iteration.
Usage

gx.md.gait.closed(xx, wts = NULL, trim = -1, mvtstart = FALSE, 
mcdstart = FALSE, main = deparse(substitute(xx)), 
ifadd = c(0.98, 0.95, 0.9), cexf = 0.6, cex = 0.8, ...)

Arguments

xx the n by p matrix for which the Mahalanobis distances are required.
wts the vector of weights for the n individuals, either zero or 1.
trim the desired trim: trim < 0 - no trim, the default; trim > 0 & < 1 - the fraction, 0 to 1 proportion, of individuals to be trimmed; trim >= 1 - the number of individuals with the highest MDs from the previous iteration to trim.
mvtstart set mvtstart = TRUE for a percentage based MVT (multivariate trim) start.
mcdstart set mcdstart = TRUE for a minimum covariance determinant (mcd) robust start.
main an alternative plot title to the default input data matrix name, see Details below.
ifadd if probability based fences are to be displayed on the Chi-square plots enter the probabilities here, see Details below. For no fences set ifadd = NULL.
cexf the scale expansion factor for the Ch-square fence annotation, by default cexf = 0.6.
cex the scale expansion factor for the symbols and text annotation within the ‘frame’ of the Chi-square plot, by default cex = 0.8.
...

Details

The variables of the input data matrix must all be expressed in the same units. An isometric log-ratio (ilr) is undertaken and the transformed data used for the GAIT process. At the completion of the process the final ilr estimates, including the inverse of the covariance matrix, are transformed to the centred log-ratio (clr) basis. The vector of means and the inverse of the covariance matrix on a clr basis are required by function gx.mvalloc.closed, that is undertaken on a clr basis.

If main is undefined the name of the matrix object passed to the function is used as the plot title. This is the recommended procedure as it helps to track the progression of the GAIT. Alternate plot titles can be defined if the final saved object is passed to gx.md.plot. If no plot title is required set main = "", or if a user defined plot title is required it may be defined, e.g., main = "Plot Title Text".

By default three fences are placed on the Chi-square plots at probabilities of membership of the current ‘core’ data subset, or total data if appropriate, with ifadd = c(0.98, 0.95, 0.9). Alternate probabilities may be defined as best for the display. If no fences are required set ifadd = NULL.

The Mahalanobis distance, Chi-square, plot x-axis label is set appropriately to indicated the type of robust start or trim using the value of proc.
Value

The following are returned as an object to be saved for the next iteration or final use:

main by default (recommended) the input data matrix name.

input the data matrix name, input = deparse(substitute(xx)), retained to be used by post-processing display functions.

matnames the row numbers and column headings of the input matrix.

proc the procedure followed for this iteration, used for subsequent Chi-square plot x-axis labelling.

wts the vector of weights for the n individuals, either zero or 1.

n the total number of individuals (observations, cases or samples) in the input data matrix.

ptrim the percentage, as a fraction, of samples called to be trimmed in this iteration, otherwise ptrim = -1.

mean the p length vector of clr basis means for the ‘core’ data following the current GAIT step.

cov the p x p clr basis covariance matrix for the ‘core’ data following the current GAIT step.

cov.inv the p x p inverse of the covariance matrix following its transformation to the clr basis from the ilr basis. For use by function gx.malloc.closed.

sd the p length vector of clr basis standard deviations for the ‘core’ data following the current GAIT step.

md the vector of Mahalanobis distances for all the n individuals following the current GAIT step.

ppm the vector of predicted probabilities of membership for all the n individuals following the current GAIT step.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data matrix, must be removed prior to executing this function, see ltdl.fix.df.

Any rows in the data matrix with NAs are removed in the ilr transformation.

Warnings are generated when the number of individuals (observations, cases or samples) falls below 5p, and additional warnings when the number of individuals falls below 3p. At these low ratios of individuals to variables the shape of the p-space hyperellipsoid is difficult to reliably define, and therefore the results may lack stability. These limits 5p and 3p are generous, the latter especially so; many statisticians would argue that the number of individuals should not fall below 9p, see Garrett (1993).

Author(s)

Robert G. Garrett
References

Garrett, R.G., 1988. IDEAS - An interactive computer graphics tool to assist the exploration geo-

Newsletter, 81:9-14.

12th International Geochemical Exploration Symposium, Geochemical Exploration 1987 (Ed. S.

See Also

ltdl.fix.df, remove.na, gx.md.plot, gx.md.print, ilr

Examples

## Make test data available
data(sind.mat2open)

## To multivariate trim as in IDEAS, see JGE (1989) 32(1-3):319-341,
## but recognizing that the data are of a closed compositional form
## and using a mcd start, execute:
gx.md.gait.closed(sind.mat2open, ifadd = 0.95)
sind.gait.1 <- gx.md.gait.closed(sind.mat2open, mcdstart = TRUE,
ifadd = NULL)
sind.gait.2 <- gx.md.gait.closed(sind.mat2open, wts = sind.gait.1$wts,
mvtstart = TRUE, trim = 3, ifadd = 0.9)
sind.gait.3 <- gx.md.gait.closed(sind.mat2open, wts = sind.gait.2$wts,
trim = 1, ifadd = 0.9)

## Display saved object with alternate main titles and list outliers
gx.md.plot(sind.gait.3, cex.main = 0.8, ifadd = 0.9,
main = "Howarth & Sinding-Larsen\nStream Sediments")
gx.md.print(sind.gait.3, pcut = 0.2)

## Clean-up
rm(sind.gait.1)
rm(sind.gait.2)
rm(sind.gait.3)

gx.md.plot Function to Display Chi-square Plots of Mahalanobis Distances

Description

Function to display Chi-square plots of Mahalanobis distances from objects saved from gx.mva,
gx.mva.closed, gx.robmva, gx.robmva.closed, gx.md.gait and gx.md.gait.closed. The ac-
tual plotting of the displays is undertaken by function gx.md.plt0. The function facilitates mak-
ing 'cosmetic' changes to the Chi-square plots not so easily achieved in function gx.md.gait
and \textit{gx md.gait.closed}, and not possible in functions \textit{gx.mva}, \textit{gx.mva.closed}, \textit{gx.robmva} or \textit{gx.robmva.closed}.

Usage

\texttt{gx.md.plot(save, main = "", ifadd = c(0.98, 0.95, 0.9), cexf = 0.6, cex = 0.8, \ldots)}

Arguments

\begin{itemize}
  \item \texttt{save} a saved object from the execution of function \textit{gx.mva}, \textit{gx.mva.closed}, \textit{gx.robmva}, \textit{gx.robmva.closed}, \textit{gx.md.gait} or \textit{gx.md.gait.closed}.
  \item \texttt{main} an alternate Chi-square plot title to that in the saved object, see Details below.
  \item \texttt{ifadd} if probability based fences are to be displayed on the Chi-square plots enter the probabilities here, see Details below. For no fences set \texttt{ifadd = NULL}.
  \item \texttt{cexf} the text scale expansion factor for the annotation of the probability based fences, by default \texttt{cexf = 0.6}.
  \item \texttt{cex} the text scale expansion factor for the other annotation within the ‘frame’ of the Chi-square plot, by default \texttt{cex = 0.8}.
  \item \texttt{\ldots} further arguments to be passed to methods concerning the generated plots. For example, if some colour other than black is required for the plotting characters, specify \texttt{col = 2} to obtain red (see \texttt{display.lty} for the default colour palette). If it is required to make the plot title or axis labelling smaller, add \texttt{cex.main = 0.9} or \texttt{cex.lab = 0.9}, respectively, to reduce the font size by 10%.
\end{itemize}

Details

If \texttt{main} is undefined the name of the matrix object from which the Mahalanobis distances were derived is passed to the function via the saved object. Using the matrix name is the recommended procedure as it helps to track the progression during a GAIT exercise, and acts as a record of the data source. However, at a presentation stage an alternate plot title may preferred and can be defined in this function, e.g., \texttt{main = "Plot Title Text"}. If no plot title is required set \texttt{main = ""}.

By default three fences are placed on the Chi-square plots at probabilities of membership of the current ‘core’ data subset, or total data if appropriate, with \texttt{ifadd = c(0.98, 0.95, 0.9)}. Alternate probabilities may be defined as best for the display. If no fences are required set \texttt{ifadd = NULL}.

The Mahalanobis distance, Chi-square, plot x-axis label is set appropriately to indicated the type of robust start or trim using the value of \texttt{proc} from the saved object passed to the function.

Author(s)

Robert G. Garrett

See Also

\texttt{gx.md.gait, gx.md.gait.closed, gx.mva, gx.mva.closed, gx.robmva, gx.robmva.closed, gx.add.chisq}
Examples

```r
## Make test data available
data(sind.mat2open)

## Save and display Chi-square plot
data.save <- gx.mva(ilr(sind.mat2open))
gx.md.plot(data.save)
gx.md.plot(data.save,
main = "Howarth & Sinding Larsen Stream Sediments\n\nilr transform",
cex = 0.8, cex = 1, col = 2)

## Save and display Chi-square plot with a
## mcd robust start and ilr transformation
data.save <- gx.md.gait(ilr(sind.mat2open), mcdstart = TRUE, mvtstart = TRUE, trim = 3, ifadd = NULL)
gx.md.plot(data.save)
gx.md.plot(data.save,
main = paste("Howarth & Sinding-Larsen\nStream Sediments, ilr Transformed Data",
"\n\nMCD robust start"), ifadd = 0.9, cex.main = 0.8)

## Clean-up
rm(data.save)
```

---

**gx.md.plt0**

*Function to Display Chi-square plots of Mahalanobis Distances*

**Description**

This function is not called directly by the user but from functions that plot Mahalanobis distances, i.e. `gx.md.gait` and `gx.md.plot`.

**Usage**

```r
gx.md.plt0(md, n, p, trim = trim, ptrim = -1, proc = proc, main = main, ifadd = ifadd, cexf = cexf, cex = cex, ...)
```

**Arguments**

- `md` a vector of Mahalanobis distances of length `n`.
- `n` the length of the vector of Mahalanobis distances.
- `p` the number of variables upon which the Mahalanobis distances are based.
- `trim` the number of individuals (observations or samples) that have been trimmed, and did not contribute to the estimation of covariance and means.
- `ptrim` the percentage trim requested, if a percentage (MVT) trim was requested.
- `proc` the procedure by which the Mahalanobis distances were estimated, used to ensure appropriate labelling of the Chi-square plot x-axis.
main the title for the Chi-square plot.
ifadd the probability based fences to be displayed on the Chi-square plots, set by the calling function and the user.
cexf the text scale expansion factor for the annotation of the probability based fences, set by the calling function and the user.
cex the text scale expansion factor for the other annotation within the ‘frame’ of the Chi-square plot, set by the calling function and the user.
... further arguments to be passed to methods concerning the generated plots. For example, if some colour other than black is required for the plotting characters, specify col = 2 to obtain red (see display.lty for the default colour palette). If it is required to make the plot title or axis labelling smaller, add cex.main = 0.9 or cex.lab = 0.9, respectively, to reduce the font size by 10%.

Author(s)
Robert G. Garrett

See Also
gx.md.gait, gx.md.plot, gx.add.chisq

Examples

## Make test data available
data(sind.mat2open)

## Generate and display sets of Mahalanobis distances
gx.md.gait(ilr(sind.mat2open))
gx.md.gait(ilr(sind.mat2open), mcdstart = TRUE, ifadd = NULL)
gx.md.gait(ilr(sind.mat2open), mcdstart = TRUE, mvtstart = TRUE, trim = 3, ifadd = 0.9)

---

**gx.md.print**

*Function to Display Membership Probabilities*

**Description**

Function to display the Mahalanobis distances (MDs) and predicted probabilities of membership (ppm) following computations by functions `gx.md.gait`, `gx.md.gait.closed`, `gx.mva`, `gx.mva.closed`, `gx.robmva` or `gx.robmva.closed`. The user may select the predicted probability of membership below which the results are displayed. Alternately the Mahalanobis distances and group membership probabilities may be saved as a `.csv` file for future use.

**Usage**

gx.md.print(save, pcut = 0.1, file = NULL)
Arguments

save
a saved object from any of functions \texttt{gx.md.gait}, \texttt{gx.md.gait.closed}, \texttt{gx.mva}, \texttt{gx.mva.closed}, \texttt{gx.robmva}, or \texttt{gx.robmva.closed}.

pcut
the probability of group membership below which records will be displayed on the current device in ascending order of membership probability, i.e. most outlying individuals first.

file
the file name for saving the function output in the R working directory, see Details below.

Details

The Mahanalobis distances, the membership probabilities and input matrix row numbers are extracted from the saved object and sorted in increasing order of probabilities of group membership for display on the current device. The full table of Mahalanobis distances and group membership probabilities may be saved as a ‘.csv’ file in the working directory, with the ‘.csv’ being appended in the function. See example below. If \texttt{file = ""} or \texttt{file = ""} a default file name is generated from the input file name from the function that generated the Mahalanobis distances and "_MDs.csv".

Value

The last displayed table, \texttt{table.rows}, is returned and may be saved as an object if required.

Note

This function is similar in purpose to \texttt{gx.mvalloc.print} for displaying multivariate outliers, however, it operates on a single population.

Author(s)

Robert G. Garrett

See Also


Examples

```r
## Make test data available
data(sind.mat2open)

## Estimate and display robust Mahalanobis distances
sind.save <- gx.md.gait.closed(sind.mat2open, mcdstart = TRUE, ifadd = NULL)
gx.md.print(sind.save, pcut = 0.3)

## Save display for future use
gx.md.print(sind.save, file = "sind.save.ilr.mcd.md")

## Clean-up
rm(sind.save)
```
Description

The function carries out a Principal Components Analysis (PCA) and estimates the Mahalanobis distances for a dataset and places them in an object to be saved and post-processed for display and further manipulation. Classical procedures are used, for robust procedures see gx.robmva. For results display see gx.rqpca.screeplot, gx.rqpca.loadplot, gx.rqpca.plot, gx.rqpca.print, gx.md.plot and gx.md.print. For Kaiser varimax rotation see gx.rotate. For closed compositional data use gx.mva.closed.

Usage

gx.mva(xx, main = deparse(substitute(xx)))

Arguments

xx a n by p data matrix to be processed.
main by default the name of the object xx, main = deparse(substitute(xx)), it may be replaced by the user, but this is not recommended, see Details below.

Details

If main is undefined the name of the matrix object passed to the function is used to identify the object. This is the recommended procedure as it helps to track the progression of a data analysis. Alternate plot titles are best defined when the saved object is passed to gx.rqpca.plot, gx.rqpca.screeplot or gx.md.plot for display. If no plot title is required set main = "", or if a user defined plot title is required it may be defined, e.g., main = "Plot Title Text".

Value

The following are returned as an object to be saved for subsequent display, etc.:

main by default (recommended) the input data matrix name.
input the data matrix name, input = deparse(substitute(xx)), retained to be used by post-processing display functions.
proc the procedure used, by default proc = "cov" to indicate a classical covariance matrix.
n the total number of individuals (observations, cases or samples) in the input data matrix.
nc the number of individuals remaining in the ‘core’ data subset after trimming. At this stage of a data analysis nc = n.
p the number of variables on which the multivariate operations were based.
ifilr flag for gx.md.plot, set to FALSE.
matnames the row numbers or identifiers and column headings of the input matrix.
wts the vector of weights for the n individuals used to compute the covariance matrix and means. At this stage of the data analysis all weights are set to ‘1’.
mean the vector the weighted means for the p variables.
cov the p by p weighted covariance matrix for the n by p data matrix.
sd the vector of weighted standard deviations for the p variables.
snd the n by p matrix of weighted standard normal deviates.
r the p by p matrix of weighted Pearson product moment correlation coefficients.
eigenvalues the vector of p eigenvalues of the scaled Pearson correlation matrix for RQ analysis, see Grunsky (2001).
econtrib the vector of p eigenvalues each expressed as a percentage of the sum of the eigenvalues.
eigenvectors the n by p matrix of eigenvectors.
rload the p by p matrix of Principal Component (PC) loadings.
rcr the p by p matrix containing the percentages of the variability of each variable (rows) expressed in each PC (columns).
rqscore the n by p matrix of the n individuals scores on the p PCs.
vcontrib a vector of p variances of the columns of rqscore.
pvcontrib the vector of p variances of the columns of rqscore expressed as percentages. This is a check on vector econtrib, the values should be identical.
cpvcontrib the vector of p cumulative sums of pvcontrib, see above.
md the vector of n Mahalanobis distances (MDs) for the n by p input matrix.
ppm the vector of n predicted probabilities of population membership, see Garrett (1990).
epm the vector of n empirical Chi-square probabilities for the MDs.
nr the number of PCs that have been rotated. At this stage of a data analysis nr = NULL in order to control PC plot axis labelling.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df. Any rows in the data matrix with NAs are removed prior to computations. In the instance of a compositional data opening transformation NAs have to be removed prior to undertaking the transformation, see na.omit.where.na and remove.na. When that procedure is followed the opening transformations may be executed on calling the function, see Examples below.

Note that, executing a clr transformation leads to a singular matrix that can not be inverted for the estimation of Mahalanobis distances. In that case the values of md, ppm and epm are all set to NULL.

Note that, executing a ilr transformation permits the estimation of Mahalanobis distances and associated probabilities through the use of p-1 synthetic variables. However, in that instance the loadings of the p-1 synthetic variables will be plotted by gx.rqpcaplot rather than the loadings for the elements.
Therefore, use function `gx.mva.closed` for compositional, geochemical, data.

Warnings are generated when the number of individuals (observations, cases or samples) falls below 5p, and additional warnings when the number of individuals falls below 3p. At these low ratios of individuals to variables the shape of the p-space hyperellipsoid is difficult to reliably define, and therefore the results may lack stability. These limits 5p and 3p are generous, the latter especially so; many statisticians would argue that the number of individuals should not fall below 9p, see Garrett (1993).

**Author(s)**

Robert G. Garrett

**References**


**See Also**

`ltdl.fix.df`, `remove.na`, `na.omit`, `gx.rqpca.screeplot`, `gx.rqpca.loadplot`, `gx.rqpca.plot`, `gx.rqpca.print`, `gx.md.plot`, `gx.md.print`, `gx.robmva`, `gx.robmva.closed`, `gx.rotate`

**Examples**

```r
## Make test data available
data(sind.mat2open)

## Generate gx.mva object, for demonstration purposes only
## These are compositional data - gx.mva.closed should be used
sind.save <- gx.mva(sind.mat2open)
gx.rqpca.screeplot(sind.save)
gx.rqpca.loadplot(sind.save)
gx.rqpca.plot(sind.save)

## Display saved object with alternate main titles
gx.rqpca.loadplot(sind.save, main = "Howarth & Sinding-Larsen\nStream Sediments, clr Transformed Data", cex.main = 0.8)
gx.rqpca.plot(sind.save, main = "Howarth & Sinding-Larsen\nStream Sediments, clr Transformed Data", cex.main = 0.8)

## Display Mahalanobis distances in a Chi-square plot
gx.md.plot(sind.save)
```
gx.mva.closed

Function to undertake an Exploratory Multivariate Data Analysis on Compositional, geochemical data

Description
The function carries out a Principal Components Analysis (PCA) and estimates the Mahalanobis distances for a compositional dataset and places them in an object to be saved and post-processed for display and further manipulation. Classical procedures are used, for robust procedures see gx.robmva.closed. For results display see gx.rqpca.screeplot, gx.rqpca.loadplot, gx.rqpca.plot, gx.rqpca.print, gx.md.plot and gx.md.print. For Kaiser varimax rotation see gx.rotate.

Usage
gx.mva.closed(xx, main = deparse(substitute(xx)))

Arguments
xx
a n by p data matrix to be processed.

main
by default the name of the object xx, main = deparse(substitute(xx)), it may be replaced by the user, but this is not recommended, see Details below.

Details
If main is undefined the name of the matrix object passed to the function is used to identify the object. This is the recommended procedure as it helps to track the progression of a data analysis. Alternate plot titles are best defined when the saved object is passed to gx.rqpca.loadplot, gx.rqpca.plot, gx.rqpca.screeplot or gx.md.plot for display. If no plot title is required set main = " ", or if a user defined plot title is required it may be defined, e.g., main = "Plot Title Text".

The data are centre log-ratio transformed prior to undertaking the PCA. For the computation of Mahalanobis distances the data are isometrically log-ratio transformed, this results in the loss of one degree of freedom.

Value
The following are returned as an object to be saved for subsequent display, etc.:

main
by default (recommended) the input data matrix name.

input
the data matrix name, input = deparse(substitute(xx)), retained to be used by post-processing display functions.
proc the procedure used, by default proc = "cov" to indicate a classical covariance matrix.

n the total number of individuals (observations, cases or samples) in the input data matrix.

nc the number of individuals remaining in the ‘core’ data subset after trimming. At this stage of a data analysis nc = n.

p the number of variables on which the multivariate operations were based.

ifilr flag for gx.md.plot, set to TRUE.

matnames the row numbers or identifiers and column headings of the input matrix.

wts the vector of weights for the n individuals used to compute the covariance matrix and means. For a classical, non-robust, estimation all weights are set to ‘1’.

mean the vector the clr means for the p variables.

cov the p by p clr covariance matrix for the n by p data matrix.

sd the vector of clr standard deviations for the p variables.

snd the n by p matrix of clr standard normal deviates.

r the p by p matrix of clr Pearson product moment correlation coefficients.

eigenvalues the vector of p eigenvalues of the scaled Pearson correlation matrix for RQ analysis, see Grunsky (2001).

econtrib the vector of p eigenvalues each expressed as a percentage of the sum of the eigenvalues.

eigenvectors the n by p matrix of eigenvectors.

rload the p by p matrix of Principal Component (PC) loadings.

rcr the p by p matrix containing the percentages of the variability of each variable (rows) expressed in each PC (columns).

rqscore the n by p matrix of the n individuals scores on the p PCs.

vcontrib a vector of p variances of the columns of rqscore.

pvcontrib the vector of p variances of the columns of rqscore expressed as percentages. This is a check on vector econtrib, the values should be identical.

cpvcontrib the vector of p cumulative sums of pvcontrib, see above.

md the vector of n Mahalanobis distances (MDs) for the n by p, now (p-1), input matrix.

ppm the vector of n predicted probabilities of population membership, see Garrett (1990).

epm the vector of n empirical Chi-square probabilities for the MDs.

nr the number of PCs that have been rotated. At this stage of a data analysis nr = NULL in order to control PC plot axis labelling.
Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes
representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df.

Any rows in the data matrix with NAs are removed prior to computations. In the instance of a composi-
tional data opening transformation NAs have to be removed prior to undertaking the transformation,
see na.omit, where.na and remove.na. When that procedure is followed the opening transforma-
tions may be executed on calling the function, see Examples below.

Warnings are generated when the number of individuals (observations, cases or samples) falls below
5p, and additional warnings when the number of individuals falls below 3p. At these low ratios of
individuals to variables the shape of the p-space hyperellipsoid is difficult to reliably define, and
therefor the results may lack stability. These limits 5p and 3p are generous, the latter especially so;
many statisticians would argue that the number of individuals should not fall below 9p, see Garrett
(1993).

Author(s)

Robert G. Garrett

References

Garrett, R.G., 1990. A robust multivariate allocation procedure with applications to geochemical
data. In Proc. Colloquium on Statistical Applications in the Earth Sciences (Eds F.P. Agterberg &

Newsletter, 81:9-14.

and R. Computers & Geosciences, 27(2):229-235.


See Also

ltdl.fix.df, remove.na, na.omit, gx.rqPCA.screeplot, gx.rqPCA.loadplot, gx.rqPCA.plot,
gx.rqPCA.print, gx.md.plot, gx.md.print, gx.robmva, gx.robmva.closed, gx.rotate

Examples

```r
## Make test data available
data(sind.mat2open)

## Generate gx.mva object after an clr transform for a PCA
sind.closed <- gx.mva.closed(sind.mat2open)
gx.rqPCA.screepplot(sind.closed)
gx.rqPCA.plot(sind.closed)
gx.rqPCA.loadplot(sind.closed)

## Display saved object with alternate main titles
gx.rqPCA.loadplot(sind.closed,
main = "Howarth & Sinding-Larsen\nStream Sediments, clr Transformed Data",
```
gx.mvalloc

Function for Allocation on the basis of Multivariate Data

Description

Function to allocate individuals (observations, cases or samples) into one of up to nine (9) reference groups (populations) on the basis of their Mahalanobis distances. If an individual’s predicted probability of group membership (typicality) falls below a user defined ‘cut-off’, \( p_{crit} \), the individual is allocated to an ‘outlier bin’.

Usage

\[
\text{gx.mvalloc}(pcrit = 0.05, x, \ldots)
\]

Arguments

- \( pcrit \): the critical cut-off probability for group membership below which an individual will be classified as an ‘outlier’. By default the critical probability of group membership is set to \( pcrit = 0.05 \).
- \( x \): a \( n \) by \( p \) matrix containing the \( n \) individuals, with \( p \) variables determined on each, to be allocated, see Details below.
- \( \ldots \): a list of objects, up to a maximum of nine (9), saved from any of functions \text{gx.md.gait}, \text{gx.md.gait.closed}, \text{gx.mva}, \text{gx.robmva} or \text{gx.robmva.closed}, containing the vectors of means and covariance matrices for the ‘reference’ groups into which the individuals are to be classified.

Details

It is imperative that the data matrix \( x \) contains no special codes and all records (individuals) with NAs have been removed, see Notes below. It is also imperative that the variables in the reference groups and in the matrix \( x \) of individuals to be classified are identical and in the same order.

The allocations are made on the assumption that the covariance structures are inhomogeneous, i.e. that the population hyperellipsoids are of different size, shape and orientation in \( p \)-space.
Value

The following are returned as an object to be saved for display with \texttt{gx.mvalloc.print}:

- \texttt{groups} a list of the names of the \texttt{kk} reference groups.
- \texttt{kk} the number of reference groups passed to the function.
- \texttt{n} the number of individuals (observations, cases or samples) allocated.
- \texttt{p} the number of variables in the reference and allocated data.
- \texttt{pcrit} the critical cut-off probability for reference group membership.
- \texttt{pgm} a vector of \texttt{kk} predicted probabilities of reference group memberships.
- \texttt{xalloc} the reference group, \texttt{1:kk}, that the individual was allocated into. All outliers, i.e. all \texttt{pgm} \texttt{1:kk} \texttt{< crit} are allocated to group zero, \texttt{0}. Therefore \texttt{xalloc} will be in the range of \texttt{0:kk}.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed from the matrix \texttt{x} prior to executing this function, see \texttt{ltdl.fix.df}. Additionally, any rows in the data matrix with NAs also must have been removed prior to computations, see \texttt{na.omit} and \texttt{remove.na}.

It is recommended that when applying this procedure to compositional data an ilr transformation be undertaken, this can be done at execution time, see Example below. This implies that the reference group means and covariance matrices must have also been estimated following an ilr transformation.

Author(s)

Robert G. Garrett

References


See Also

\texttt{gx.md.gait}, \texttt{gx.md.gait.closed}, \texttt{gx.mva}, \texttt{gx.robmva}, \texttt{gx.robmva.closed}, \texttt{gx.mvalloc.print}, \texttt{ltdl.fix.df}, \texttt{remove.na}, \texttt{na.omit}

Examples

```r
## Generate three groups of synthetic bivariate normal data
grp1 <- mvrnorm(100, mu = c(40, 30), Sigma = matrix(c(6, 3, 3, 2, 2)))
gr1 <- cbind(grp1, rep(1, 100))
gr2 <- mvrnorm(100, mu = c(50, 40), Sigma = matrix(c(4, -3, -3, 5, 2, 2)))
gr2 <- cbind(grp2, rep(2, 100))
```
Function to allocate individuals (observations, cases or samples) from closed compositional, geo-chemical, data sets into one of up to nine (9) reference groups (populations) on the basis of their Mahalanobis distances. If an individual’s predicted probability of group membership (typicality) falls below a user defined ‘cut-off’, pcrit, the individual is allocated to an ‘outlier bin’.

```r
# Generate a set of six (6) outliers
anom <- matrix(c(35, 40, 25, 60, 25, 60, 35, 40, 25, 60, 60, 25, 60), 6, 2)
anom <- cbind(anom, rep(4, 6))

# Bind the test data sets together and display the test data
anom <- cbind(anom, rep(4, 6))
test.mvalloc <- as.data.frame(test.mvalloc.mat)
dimnames(test.mvalloc)[[2]] <- c("x", "y", "grp")
attach(test.mvalloc)

# Generate robust reference groups
test.save.grp1 <- gx.md.gait(grp1[, -3], mcdstart = TRUE)
test.save.grp2 <- gx.md.gait(grp2[, -3], mcdstart = TRUE)
test.save.grp3 <- gx.md.gait(grp3[, -3], mcdstart = TRUE)

# Allocate the synthetic data into the three reference groups
test.mvalloc <- gx.mvalloc(pcrit = 0.05, test.mvalloc.mat[, -3], test.save.grp1, test.save.grp2, test.save.grp3)

# Display the results of the allocation
xyplot.tags(x, y, test.mvalloc$xalloc, cex = 0.75)
gx.mvalloc.print(test.mvalloc)

# Save the allocation as a csv file
ngx.mvalloc.print(test.mvalloc, ifprint = FALSE, file = "test.mvalloc")

# Clean-up and detach synthetic test data
rm(grp1)
rm(grp2)
rm(grp3)
rm(anom)
rm(test.mvalloc)
rm(test.save.grp1)
rm(test.save.grp2)
rm(test.save.grp3)
rm(test.save.mvalloc)
detach(test.mvalloc)
```

### Description

Function for Allocation on the basis of Multivariate Data for closed Compositional Data
Usage

gx.mvalloc.closed(pcrit = 0.05, xx, ...)

Arguments

pcrit  the critical cut-off probability for group membership below which an individual will be classified as an 'outlier'. By default the critical probability of group membership is set to pcrit = 0.05.

xx  a n by p matrix containing the n individuals, with p variables determined on each, to be allocated, see Details below.

...  a list of objects, up to a maximum of nine (9), saved from either function gx.md.gait.closed or gx.robmva.closed, containing the vectors of means and inverse covariance matrices for the 'reference' groups into which the individuals are to be classified.

Details

It is imperative that the data matrix xx contains no special codes, see Note below. It is also imperative that the variables in the reference groups and in the matrix x of individuals to be classified are identical in number and in the same order.

The allocations are made on the assumption that the covariance structures are inhomogeneous, i.e. that the population hyperellipsoids are of different size, shape and orientation in p-space.

Value

The following are returned as an object to be saved for display with gx.mvalloc.print:

groups  a list of the names of the kk reference groups.

kk  the number of reference groups passed to the function.

n  the number of individuals (observations, cases or samples) allocated.

p  the number of variables in the reference and allocated data.

pcrit  the critical cut-off probability for reference group membership.

pgm  a vector of kk predicted probabilities of reference group memberships.

xalloc  the reference group, 1:kk, that the individual was allocated into. All outliers, i.e. all pgm(1:kk) < crit are allocated to group zero, 0. Therefore xalloc will be in the range of 0:kk.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed from the matrix xx prior to executing this function, see ltdl.fix.df. Any rows in the input data matrix xx with NAs are removed prior to computations.

Author(s)

Robert G. Garrett
References


See Also

gx.md.gait.closed, gx.robmva.closed, gx.mvalloc.print, ltdl.fix.df, remove.na, na.omit

Examples

## Make test data available
data(ogrady)
attach(ogrady)
ogrady.grdr <- gx.subset(ogrady, Lith == "GRDR")
ogrady.grnt <- gx.subset(ogrady, Lith == "GRNT")
## Ensure all data are in the same units (mg/kg)
ogrady.grdr.2open <- ogrady.grdr[, c(5:14)]
ogrady.grnt.2open <- ogrady.grnt[, c(5:14)]
ogrady.grnt.2open[, 1:7] <- ogrady.grnt.2open[, 1:7] * 10000
ogrady.2open <- ogrady[, c(5:14)]
ogrady.2open[, 1:7] <- ogrady.2open[, 1:7] * 10000

## Create reference data sets
ogrady.grdr.save <- gx.md.gait.closed(as.matrix(ogrady.grdr.2open),
mcdstart = TRUE)
ogrady.grnt.save <- gx.md.gait.closed(as.matrix(ogrady.grnt.2open),
mcdstart = TRUE)

## Allocate all O'Grady granitoids
ogrady.mvalloc <- gx.mvalloc.closed(xcrit = 0.02, as.matrix(ogrady.2open),
ogrady.grdr.save, ogrady.grnt.save)

## Display list of outliers
gx.mvalloc.print(ogrady.mvalloc)

## Display allocations
ogrady.mvalloc$xalloc

## Save allocations as a csv file
gx.mvalloc.print(ogrady.mvalloc, ifprint = FALSE,
file = "ogrady.gait.closed.mcd.mvalloc")

## Clean-up and detach test data
rm(ogrady.grdr)
rm(ogrady.grnt)
rm(ogrady.grdr.2open)
rm(ogrady.grnt.2open)
gx.mvalloc.print

Function to display the results of Multivariate Allocation

Description

Function to extract and display the results from the saved object from `gx.mvalloc` or `gx.mvalloc.closed`. The function displays on the current device only those individuals (observations, cases or samples) whose predicted probability of reference group membership was less than the value provided, pcrit, for all reference groups, i.e. the outliers. Alternately, the results can be saved as a `.csv` file for viewing with a spreadsheet program and any subsequent post-processing.

Usage

```r
gx.mvalloc.print(save, ifprint = TRUE, unalloc = TRUE, file = NULL)
```

Arguments

- **save**: an object saved from `gx.mvalloc`.
- **ifprint**: by default the ‘outliers’, i.e. individuals classified into group ‘zero’, are displayed on the current device. The display consists of the input matrix row numbers together with the predicted probabilities of reference group membership for the 1:kk reference groups. To suppress the display set `ifprint = FALSE`.
- **unalloc**: by default, `unalloc = TRUE`, individuals that were not allocated to one of the reference groups are displayed. To suppress displaying these individuals, set `unalloc = FALSE`.
- **file**: the name of the `.csv` file for allocation outcomes for the total data set to be saved in the working directory. Note, the `.csv` extension is appended in the function. See Example below.

Note

Included in the display on the current device are the names of the kk reference group objects supplied to `gx.mvalloc` together with the value of pcrit.

If `file = ""` or `file = " "`, a default file name of “mvalloc.csv” is generated.

Author(s)

Robert G. Garrett

See Also

`gx.mvalloc`, `gx.mvalloc.closed`
Examples

```r
## Make test data available
data(ogrady)
attach(ogrady)
ogrady.grdr <- gx.subset(ogrady, Lith == "GRDR")
ogrady.grnt <- gx.subset(ogrady, Lith == "GRNT")

## Ensure all data are in the same units (mg/kg)
ogrady.grdr.2open <- ogrady.grdr[, c(5:14)]
ogrady.grnt.2open <- ogrady.grnt[, c(5:14)]
ogrady.grnt.2open[, 1:7] <- ogrady.grnt.2open[, 1:7] * 10000
ogrady.2open <- ogrady[, c(5:14)]
ogrady.2open[, 1:7] <- ogrady.2open[, 1:7] * 10000

## Create reference data sets
ogrady.grdr.save <- gx.md.gait(ilr(as.matrix(ogrady.grdr.2open)),
mcdstart = TRUE)
ogrady.grnt.save <- gx.md.gait(ilr(as.matrix(ogrady.grnt.2open)),
mcdstart = TRUE)

## Allocate all O'Grady granitoids
ogrady.mvalloc <- gx.mvalloc(pcrit = 0.02, ilr(as.matrix(ogrady.2open)),
ogrady.grdr.save, ogrady.grnt.save)

## Display list of outliers
gx.mvalloc.print(ogrady.mvalloc)

## Save allocations as a csv file
gx.mvalloc.print(ogrady.mvalloc, ifprint = FALSE, file = "ogrady.mvalloc.print")

## Clean-up and detach test data
rm(ogrady.grdr)
rm(ogrady.grnt)
rm(ogrady.grdr.2open)
rm(ogrady.grnt.2open)
rm(ogrady.2open)
rm(ogrady.grdr.save)
rm(ogrady.grnt.save)
rm(ogrady.mvalloc)
detach(ogrady)
```

---

gx.pairs4parts

Display a Graphical Matrix for Parts of a Compositional Data Set

Description

Displays a graphical matrix of log10 scaled x-y plots in the upper triangle and boxplots of the ilr transforms in the lower triangle for the parts of a compositional matrix. The robust ilr stability (Filzmoser et al., 2010) for each x-y pair is displayed as the boxplot title.
Usage

gx.pairs4parts(xx, cex = 2, ifwarn = TRUE, ...)

Arguments

xx a matrix, or sub-matrix, of parts from a compositional data set.
cex by default the size of the text of the variable names in the diagonal of the graphical matrix. By default cex = 2, and may be changed if required.
ifwarn by default ifwarn = TRUE which generates a reminder/warning that when carrying out analyses of compositional data all data must be in the same measurement units. The message can be suppressed by setting ifwarn = FALSE.

... further arguments to be passed to plot or bxplo. For example, the size of the axis scale annotation can be change by setting cex.axis and the size of the plot title by setting cex.main. For example, if it is required to make the plot title smaller, add cex.main = 0.9 to reduce the font size by 10%.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df. Any data vectors (rows) containing NAs are removed prior to computation.

Author(s)

Robert G. Garrett

References


See Also

ltdl.fix.df, remove.na, bxplo, ilr.stab

Examples

## Make test data available
data(sind.mat2open)

## Display 'pairs' plots for a set, or sub-set, of parts of a
## compositional data matrix
gx.pairs4parts(sind.mat2open)
Display Pearson Correlation Coefficients and their Significances

Description

The function computes Pearson product moment correlation coefficients and places them in the upper triangle of a printed matrix displayed on the current device, the probabilities that the coefficients are not due to chance (Ho: Coefficient = 0) are printed in the lower triangle. The diagonal is filled with NAs to visually split the two triangles.

Usage

```
gx.pearson(xx, log = FALSE, ifclr = FALSE, ifwarn = TRUE)
```

Arguments

- `xx`: a matrix of numeric data.
- `log`: if `log = TRUE` the data are log10 transformed prior to computation of the Pearson coefficients. The default is no transformation.
- `ifclr`: if `ifclr = TRUE` the data are Centred Log-Ratio transformed prior to the computation of the Pearson Coefficients. The default is no transformation.
- `ifwarn`: by default `ifwarn = TRUE` which generates a reminder/warning that when carrying out a centred log-ratio transformation all the data must be in the same measurement units. The message can be suppressed by setting `ifwarn = FALSE`.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`. Any data vectors (rows) containing NAs are removed prior to computation.

This function is not recommended for use with closed compositional data sets, i.e. geochemical analyses, unless correlations are sought between a non-compositional variable and individual compositional variables. If it is used with compositional data, it is highly recommended that `ifclr` be set to `TRUE` to remove the effects of closure and display the 'true' inter-element variability. However, different groups of elements, subsets, of a data set will yield different inter-element correlations for the same pair of elements due to the nature of the clr transform. When carrying out a centred log-ratio transformation it is essential that the data are all in the same measurement units, and by default a reminder/warning is display if the data are centred log-ratio transformed, see `ifwarn` above.

For working with compositional data sets functions `gx.vm` and `gx.sm` are recommended. For visual displays see `gx.pairs4parts` and `gx.plot2parts`.

When a centred log-ratio transformation is undertaken the log 'switch' is ignored.

Author(s)

Robert G. Garrett
gx.plot2parts

Display Plots for Two Parts from a Compositional Data Set

Description
Displays a panel of four plots for a pair of parts from a compositional data set. The displays consist of a log10 scaled x-y plot, a boxplot of the corresponding values of ilr(x,y) annotated with the robust ilr stability measure, and sequential index and ECDF plots of the ilr values. The display is based on those used in Filzmoser et al. (2010).

Usage
gx.plot2parts(xx1, xx2, x1lab = deparse(substitute(xx1)), x2lab = deparse(substitute(xx2)), cex = 0.8, ifwarn = TRUE, ...)

Arguments
xx1  a column vector from a matrix or data frame of compositional data, xx1[1], ..., xx1[n].
xx2  another column vector from the matrix or data frame of compositional data, xx2[1], ..., xx2[n]. xx1 and xx2 must be of identical length, n.
x1lab the x-axis title, by default the variable name, deparse(substitute(xx1)). It is often desirable to replace this with a more informative title, e.g., x1lab = "Cu (mg/kg) in stream sediment".
x2lab the y-axis title, by default the variable name, deparse(substitute(xx2)). It is often desirable to replace this with a more informative title, e.g., x2lab = "Zn (mg/kg) in stream sediment".

Examples

## Make test data available
data(sind.mat2open)

## Compute Pearson correlation coefficients
gx.pearson(sind.mat2open)

## Compute Pearson correlation coefficients following
## a logarithmic transformation
gx.pearson(sind.mat2open, log = TRUE)

## Compute Pearson correlation coefficients following
## a centred log-ratio transformation
gx.pearson(sind.mat2open, ifclr = TRUE)

See Also
ltdl.fix.df, remove.na, clr, sind.mat2open
by default the size of the text for data set size, N, and the robust ilr stability is set to 80%, i.e. cex = 0.8, and may be changed if required.

by default ifwarn = TRUE which generates a reminder/warning that when carrying out analyses of compositional data all data must be in the same measurement units. The message can be suppressed by setting ifwarn = FALSE.

... further arguments to be passed to methods. For example, the size of the axis scale annotation can be change by setting cex.axis, the size of the axis titles by setting cex.lab, and the size of the plot title by setting cex.main. For example, if it is required to make the plot title smaller, add cex.main = 0.9 to reduce the font size by 10%.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df. Any data vectors (rows) containing NAs are removed prior to computation.

Author(s)

Robert G. Garrett

References


See Also

ltdl.fix.df, remove.na, bxplot, gx.ecdf

Examples

```r
## Make test data available
data(sind)
attach(sind)

## Display
gx.plot2parts(Cu, Zn)

## Display with alternate xy-plot titling
gx.plot2parts(Cu, Zn, x1lab = "Cu (mg/kg) in stream sediment",
             x2lab = "Zn (mg/kg) in stream sediment")

## Detach test data
detach(sind)
```
Estimate the Quantile for a Specified Fractile

**Description**

Estimates and displays the quantile for a specified fractile of a data set by linear interpolation from the ranked data. If the function is run as `temp <- gx.quantile(xx, f)` the quantile is not displayed, but retained in `temp` for subsequent use or display.

**Usage**

```r
gx.quantile(xx, f, display = TRUE)
```

**Arguments**

- `xx` : the data set for which the quantile is to be estimated.
- `f` : the fractile for which the quantile is required, must be in the range zero to 1.
- `display` : the default is to display the fractile and estimated quantile on the current device. If no display is required, set `display = FALSE`.

**Value**

- `q` : the estimated quantile.

**Note**

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Any `NA`s in the data vector are removed prior to preparing the boxplots.

The result is an approximation, and the result from the `quantile` function will likely differ by some small amount.

**Author(s)**

Based on a script shared on S-News by Nick Ellis, April 2002

**See Also**

- `ltdl.fix.df`, `remove.na`, `gx.fractile`
Examples

```r
## Make test data available
data(kola.o)
attach(kola.o)

## Estimate the 80th percentile, f = 0.8
gx.quantile(As, 0.8)
temp <- gx.quantile(As, 0.8)
temp

## Clean-up and detach test data
rm(temp)
detach(kola.o)
```

---

**gx.rma**  
*Estimate the Coefficients of the Reduced Major Axis*

**Description**

Function to estimate the coefficients and their standard errors of the Reduced Major Axis, the case of orthogonal regression, and also known as total least squares or errors in variables regression. The procedure is based on the methodology described in Miller and Kahn (1962).

**Usage**

```r
gx.rma(xx1, xx2, x1lab = deparse(substitute(xx1)),
      x2lab = deparse(substitute(xx2)), log = FALSE)
```

**Arguments**

- `xx1` the name of the first independent variable.
- `xx2` the name of the second independent variable.
- `log` if a logarithmic transformation (base 10) of the data is required to meet homogeneity of variance considerations (i.e., severe heteroscedasticity) set `log = TRUE`. This is also advisable if the range of the observations exceeds 1.5 orders of magnitude.
- `x1lab` a title for the first independent variable, the default is the variable name, `deparse(substitute(xx1))`. It is often desirable to replace the default title of the input variable name text string with a more informative title, e.g., `x1lab = "Magnetic Susceptibility - Measurement 1"`.
- `x2lab` a title for the second independent variable, the default is the variable name, `deparse(substitute(xx2))`. It is often desirable to replace the default title of the input variable name text string with a more informative title, e.g., `x2lab = "Magnetic Susceptibility - Measurement 2"`. 
Value

A list comprising of:

- `alen`: the data set size.
- `mean`: a two-element vector with the means of `x1` and `x2`.
- `sd`: a two-element vector with the standard deviations of `x1` and `x2`.
- `corr`: the Pearson correlation coefficient for `x1` and `x2`.
- `a0`: the intercept of the reduced major axis.
- `a1`: the slope of the reduced major axis.
- `sea0`: the standard error of the intercept estimate.
- `sea1`: the standard error of the slope estimate.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Any data pairs, `xx1, xx2`, containing any NAs are omitted from the calculations.

If a log transformation is undertaken and any less than or equal to zero values occur in the data the function will halt with a warning to that effect.

The coefficients may be used to plot the RMA on a x-y plot of the two measures, see example below.

Author(s)

Robert G. Garrett

References


See Also

`ltdl.fix.df, remove.na`

Examples

```r
## Make test available
data(ms.data1)
attach(ms.data1)

## Estimate RMA coefficients for duplicate measurements on rock samples
gx.rma(MS.1, MS.2, log = TRUE,
x1lab = "MS - 1", x2lab = "MS - 2")

## Display an x-y plot of the data and the RMA, ensuring a
## square plot with similar x- and y-axis labelling and
## appropriate axis labelling
```
save.rma <- gx.rma(MS.1, MS.2, log = TRUE, 
x1lab = "MS - 1", x2lab = "MS - 2")
oldpar <- par()
par(pty = "s", pch = 3)
plot(MS.1, MS.2, log = "xy", xlim = c(min(MS.1, MS.2), max(MS.1, MS.2)), 
ylim = c(min(MS.1, MS.2), max(MS.1, MS.2)), 

xlab = "Magnetic Susceptibility - Measurement 1", 
ylab = "Magnetic Susceptibility - Measurement 2")
abline(save.rma$a0, save.rma$a1, lty = 3)
par <- oldpar

## Clean-up and detach test data
rm(save.rma)
detach(ms.data1)


---

**gx.robmva**

*Function to undertake a Robust Exploratory Multivariate Data Analysis*

**Description**

The function carries out a robust Principal Components Analysis (PCA) and estimates the Mahalanobis distances for a non-compositional dataset and places them in an object to be saved and post-processed for display and further manipulation. For closed compositional, geochemical, data use function `gx.robmva.closed`. Robust procedures are used, ‘MCD’, ‘MVE’ or user supplied weights, for classical procedures see `gx.mva`. For results display see `gx.rqpca.screeplot`, `gx.rqpca.loadplot`, `gx.rqpca.plot`, `gx.rqpca.print`, `gx.md.plot` and `gx.md.print`. For Kaiser varimax rotation see `gx.rotate`.

**Usage**

```r

  gx.robmva(xx, proc = "mcd", wts = NULL, main = deparse(substitute(xx)))
```

**Arguments**

- `xx` : a n by p data matrix to be processed.
- `proc` : by default `proc = "mcd"` for the Minimum Covariance Determinant (MCD) robust procedure. Setting `proc = "mve"` results in the Minimum Volume Ellipsoid (MVE) procedure being used. If `p > 50` the MVE procedure is used. See `wts` below.
- `wts` : by default `wts = NULL` and the MCD or MVE estimation procedures will be used. If, however, a vector of `n` zero or 1 weights are supplied these will be used for robust estimation and the value of `proc` ignored.
- `main` : by default the name of the object `xx`, `main = deparse(substitute(xx))`, it may be replaced by the user, but this is not recommended, see Details below.
Details

If main is undefined the name of the matrix object passed to the function is used to identify the object. This is the recommended procedure as it helps to track the progression of a data analysis. Alternate plot titles are best defined when the saved object is passed to `gx.rq pca.screeplot`, `gx.rq pca.loadplot`, `gx.rq pca.plot` or `gx.md.plot` for display. If no plot title is required set `main = ""`, or if a user defined plot title is required it may be defined, e.g., `main = "Plot Title Text"`. The variances of the robust Principal Component scores are displayed, in a non-robust PCA these decrease with increasing component rank. However, in a robust PCA this may not be the case, and lower-order scores with high variances are often worthy of further inspection.

Value

The following are returned as an object to be saved for subsequent display, etc.:

- **main**: by default (recommended) the input data matrix name.
- **input**: the data matrix name, `input = deparse(substitute(xx))`, retained to be used by post-processing display functions.
- **proc**: the robust procedure used, the value of `proc` will be "mcd", "mve" or "wts".
- **n**: the total number of individuals (observations, cases or samples) in the input data matrix.
- **nc**: the number of individuals remaining in the ‘core’ data subset following the robust estimation, i.e. the sum of those individuals with `wts = 1`.
- **p**: the number of variables on which the multivariate operations were based.
- **ifilr**: flag for `gx.md.plot`, set to FALSE.
- **matnames**: the row numbers or identifiers and column headings of the input matrix.
- **wts**: the vector of weights for the `n` individuals arising from the robust estimation of the covariance matrix and means.
- **mean**: the length `p` vector the weighted means for the variables.
- **cov**: the `p` by `p` weighted covariance matrix for the `n` by `p` data matrix.
- **sd**: the length `p` vector of weighted standard deviations for the variables.
- **snd**: the `n` by `p` matrix of weighted standard normal deviates.
- **r**: the `p` by `p` matrix of weighted Pearson product moment correlation coefficients.
- **eigenvalues**: the vector of `p` eigenvalues of the scaled Pearson robust correlation matrix for RQ analysis, see Grunsky (2001).
- **econtrib**: the vector of `p` robustly estimated eigenvalues each expressed as a percentage of the sum of the eigenvalues.
- **eigenvectors**: the `n` by `p` matrix of robustly estimated eigenvectors.
- **rload**: the `p` by `p` matrix of robust Principal Component (PC) loadings.
- **rcr**: the `p` by `p` matrix containing the percentages of the variability of each variable (rows) expressed in each robust PC (columns).
- **rqscore**: then by `p` matrix of the `n` individuals scores on the `p` robust PCs.
- **vcontrib**: a vector of `p` variances of the columns of `rqscore`. 
pvcontrib the vector of \( p \) variances of the columns of \( rqscore \) expressed as percentages. This is a check on vector \( econtrib \), the values should be identical for a classical PCA. However, for robust PCAs this is not so as the trimmed individuals from the robust estimation have been re-introduced. As a consequence \( pvcontrib \) can be very different from \( econtrib \). The plotting of PCs containing high proportions of the variance in robust PCAs can be useful for identifying outliers.

\[
\text{cpvcontrib} \quad \text{the vector of } p \text{ cumulative sums of } pvcontrib, \text{ see above.}
\]

\[
\text{md} \quad \text{the vector of } n \text{ robust Mahalanobis distances (MDs) for the } n \times p \text{ input matrix.}
\]

\[
\text{ppm} \quad \text{the vector of } n \text{ robust predicted probabilities of population membership, see } Garrett \ (1990).
\]

\[
\text{epm} \quad \text{the vector of } n \text{ robust empirical Chi-square probabilities for the MDs.}
\]

\[
\text{nr} \quad \text{the number of PCs that have been rotated. At this stage of a data analysis } nr = \text{NULL in order to control PC plot axis labelling.}
\]

**Note**

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see \( ltdl\_fix\_df \).

Any rows in the data matrix with \( \text{NAs} \) are removed prior to computations. In the instance of a compositional data opening transformation \( \text{NAs} \) have to be removed prior to undertaking the transformation, see \( \text{na.omit, where.na and remove.na} \). When that procedure is followed the opening transformations may be executed on calling the function, see Examples below.

Passing a set of weights from an investigation with \( gx\_md\_gait \) or on the basis of some prior knowledge permits the use of a \( \text{clr} \) transformation. In this instance a Moore-Penrose inverse is computed and used for the estimation of Mahalanobis distances. See example below. With reference to weights based on prior knowledge, the weights are not necessarily constrained to be ‘0’ or ‘1’, intermediate values may be employed.

Executing a \( \text{clr} \) transformation leads to both collinearity and singularity such that neither a PCA can be undertaken or Mahalanobis distances be estimated. The function fails - do not use with a \( \text{clr} \) transformation.

Executing a \( \text{ilr} \) transformation permits the estimation of both Principal Components and Mahalanobis distances and associated probabilities through the use of \( (p-1) \) synthetic variables. However, in that instance the loadings of the \( (p-1) \) synthetic variables will be plotted by \( gx\_rqpca\_plot \) rather than the loadings for the elements.

Warnings are generated when the number of individuals (observations, cases or samples) falls below \( 5^p \), and additional warnings when the number of individuals falls below \( 3^p \). At these low ratios of individuals to variables the shape of the \( p \)-space hyperellipsoid is difficult to reliably define, and therefore the results may lack stability. These limits \( 5^p \) and \( 3^p \) are generous, the latter especially so; many statisticians would argue that the number of individuals should not fall below \( 9^p \), see Garrett (1993).

**Author(s)**

Robert G. Garrett
References


See Also

ldtl.fix.df, remove.na, na.omit, gx.rqpca.screeplot, gx.rqpca.loadplot, gx.rqpca.plot, gx.rqpca.print, gx.md.plot, gx.md.print, gx.robmva.closed, gx.rotate

Examples

### Generate a population of synthetic bivariate normal data
gr1 <- mvrnorm(100, mu = c(40, 30), Sigma = matrix(c(6, 3, 3, 2), 2, 2))
gr1 <- cbind(gr1, rep(1, 100))

### Generate a set of six (6) outliers
anom <- matrix(c(43, 34, 50, 37, 47, 30, 27, 29, 35, 33, 32, 25), 6, 2)
anom <- cbind(anom, rep(2, 6))

### Bind the test data together and display the test data
test.robmva.mat <- rbind(gr1, anom)
test.robmva <- as.data.frame(test.robmva.mat)
dimnames(test.robmva)[[2]] <- c("x", "y", "grp")
attach(test.robmva)
xyplot(tags(x, y, dimnames(test.robmva)[[1]]), cex = 0.75)

### Generate gx.robmva saved object
save.rob <- gx.robmva(as.matrix(test.robmva[, c(1:2)]))

### Display saved object with alternate main titles
gx.rqpca.screeplot(save.rob, main = "Bivariate synthetic data")
gx.rqpca.plot(save.rob, cex.lab = 0.8, rowids = TRUE, main = "Bivariate synthetic data")
gx.md.plot(save.rob, cex.main = 0.9, cex.lab = 0.8, cex.axis = 0.8, main = "Bivariate synthetic data")
gx.md.print(save.rob, pcut = 0.05)

### Clean-up and detach test data
rm(gr1)
rm(anom)
rm(test.robmva.mat)
rm(test.robmva)
rm(save.rob)
detach(test.robmva)
Function to undertake a Robust Closed Data Multivariate EDA

Description

The function carries out a robust Principal Components Analysis (PCA) and estimates the Mahalanobis distances for a closed compositional, geochemical, dataset and places the results in an object to be saved and post-processed for display and further manipulation. Robust procedures are used, ‘MCD’, ‘MVE’ or user supplied weights, for classical procedures see `gx.mva.closed`, or for non-compositional data and robust procedures see `gx.robmva`. For results display see `gx.rqPCA.screeplot`, `gx.rqPCA.loadplot`, `gx.rqPCA.plot`, `gx.rqPCA.print`, `gx.md.plot` and `gx.md.print`. For Kaiser varimax rotation see `gx.rotate`.

Usage

`gx.robmva.closed(xx, proc = "mcd", wts = NULL, main = deparse(substitute(xx)))`

Arguments

- **xx**: a n by p data matrix to be processed.
- **proc**: by default `proc = "mcd"` for the Minimum Covariance Determinant (MCD) robust procedure. Setting `proc = "mve"` results in the Minimum Volume Ellipsoid (MVE) procedure being used. If p > 50 the MVE procedure is used. See wts below.
- **wts**: by default `wts = NULL` and the MCD or MVE estimation procedures will be used. If, however, a vector of n zero or 1 weights are supplied these will be used for robust estimation and the value of `proc` ignored.
- **main**: by default the name of the object `xx`, `main = deparse(substitute(xx))`, it may be replaced by the user, but this is not recommended, see Details below.

Details

The data are initially isometrically log-ratio transformed and a robust covariance matrix and vector of means estimated, by either the Minimum Covariance Determinant (MCD) or Minimum Volume Ellipsoid (MVE) procedures, or on the basis of a vector of user supplied weights. The Mahalanobis distances are computed on the basis of the ilr transformed data. The ilr transformed data and robust estimates, including the inverse of the covariance matrix, are then back-transformed to the centred log-ratio basis and a Principal Components Analysis (PCA) undertaken (see Filzmoser, et al., 2009), permitting the results to be interpreted in the original variable space.

The variances of the robust Principal Component scores are displayed, in a non-robust PCA these decrease with increasing component rank. However, in a robust PCA this may not be the case, and lower-order scores with high variances are often worthy of further inspection.

If `main` is undefined the name of the matrix object passed to the function is used to identify the object. This is the recommended procedure as it helps to track the progression of a data analysis. Alternate plot titles are best defined when the saved object is passed to `gx.rqPCA.plot`. 
gx.robmva.closed

**gx.rpqca.screeplot** or **gx.md.plot** for display. If no plot title is required set `main = " "`, or if a user defined plot title is required it may be defined, e.g., `main = "Plot Title Text"`.

**Value**

The following are returned as an object to be saved for subsequent display, etc.:

- **main**
  - by default (recommended) the input data matrix name.

- **input**
  - the data matrix name, `input = deparse(substitute(xx))`, retained to be used by post-processing display functions.

- **proc**
  - the robust procedure used, the value of `proc` will be "mcd", "mve" or "wts".

- **n**
  - the total number of individuals (observations, cases or samples) in the input data matrix.

- **nc**
  - the number of individuals remaining in the ‘core’ data subset following the robust estimation, i.e. the sum of those individuals with `wts = 1`.

- **p**
  - the number of variables on which the multivariate operations were based.

- **ifilr**
  - flag for **gx.md.plot**, set to TRUE.

- **matnames**
  - the row numbers or identifiers and column headings of the input matrix.

- **wts**
  - the vector of weights for the `n` individuals arising from the robust estimation of the covariance matrix and means.

- **mean**
  - the length `p` vector of clr-based weighted means for the variables.

- **cov**
  - the `p` by `p` weighted clr-based covariance matrix for the `n` by `p` data matrix.

- **cov.inv**
  - the `p` by `p` weighted clr-based inverse of the covariance matrix, for use by function **gx.mvalloc.closed**.

- **sd**
  - the length `p` vector of weighted clr-based standard deviations for the variables.

- **snd**
  - the `n` by `p` matrix of clr-based weighted standard normal deviates.

- **r**
  - the `p` by `p` matrix of weighted clr-based Pearson product moment correlation coefficients.

- **eigenvalues**
  - the vector of `p` eigenvalues of the scaled clr-based Pearson robust correlation matrix for RQ analysis, see Grunsky (2001).

- **econtrib**
  - the vector of `p` robustly estimated eigenvalues each expressed as a percentage of the sum of the eigenvalues.

- **eigenvectors**
  - the `n` by `p` matrix of clr-based robustly estimated eigenvectors.

- **rload**
  - the `p` by `p` matrix of robust clr-based Principal Component (PC) loadings.

- **rcr**
  - the `p` by `p` matrix containing the percentages of the variability of each variable (rows) expressed in each robust clr-based PC (columns).

- **rqscore**
  - the `n` by `p` matrix of the `n` individuals scores on the `p` robust clr-based PCs.

- **vcontrib**
  - a vector of `p` variances of the columns of **rqscore**.

- **pvcontrib**
  - the vector of `p` variances of the columns of **rqscore** expressed as percentages. This is a check on vector **econtrib**, the values should be identical for a classical PCA. However, for robust PCAs this is not so as the trimmed individuals from the robust estimation have been re-introduced. As a consequence **pvcontrib** can be very different from **econtrib**. The plotting of PCs containing high proportions of the variance in robust PCAs can be useful for identifying outliers.
The vector of p cumulative sums of pvcontrib, see above.

The vector of n robust ilr-based Mahalanobis distances (MDs) for the n by p input matrix.

The vector of n robust ilr-based predicted probabilities of population membership, see Garrett (1990).

The vector of n robust ilr-based empirical Chi-square probabilities for the MDs.

The number of PCs that have been rotated. At this stage of a data analysis nr = NULL in order to control PC plot axis labelling.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df.

Any rows in the data matrix with NAs are removed prior to computations. In the instance of a compositional data opening transformation NAs have to be removed prior to undertaking the transformation, see na.omit, where.na and remove.na. When that procedure is followed the opening transformations may be executed on calling the function, see Examples below.

Warnings are generated when the number of individuals (observations, cases or samples) falls below 5*p, and additional warnings when the number of individuals falls below 3*p. At these low ratios of individuals to variables the shape of the p-space hyperellipsoid is difficult to reliably define, and therefore the results may lack stability. These limits 5*p and 3*p are generous, the latter especially so; many statisticians would argue that the number of individuals should not fall below 9*p, see Garrett (1993).

Author(s)

Robert G. Garrett

References


See Also

ltdl.fix.df, remove.na, na.omit, orthonorm, gx.rqPCA.screeplot, gx.rqPCA.loadplot, gx.rqPCA.plot, gx.rqPCA.print, gx.md.plot, gx.md.print, gx.robmva, gx.rotate
gx.rotate

**Examples**

```r
evaln(data(sind.mat2open))
evaln(sind.save <- gx.robmva.closed(sind.mat2open))
evaln(gx.md.plot(sind.save))
evaln(gx.rqpca.screeplot(sind.save))
evaln(gx.rqpca.loadplot(sind.save))

## Display appropriately annotated results

evaln(gx.md.plot(sind.save, main = "Howarth & Sinding-Larsen
Stream Sediments, Opened Data", cex.main=0.8))
evaln(gx.rqpca.screeplot(sind.save, main = "Howarth & Sinding-Larsen Stream Sediments
Open Data")
evaln(gx.rqpca.plot(sind.save, rowids = TRUE, main = "Howarth & Sinding-Larsen Stream Sediments
Open Data")
evaln(sind.save$pvcontrib)
evaln(gx.rqpca.plot(sind.save, v1 = 3, v2 =4, rowids = TRUE, main = "Howarth & Sinding-Larsen Stream Sediments
Open Data")

## Display Kaiser Varimax rotated (nrot = 4) results

evaln(sind.save.rot4 <- gx.rotate(sind.save, 4))
evaln(gx.rqpca.plot(sind.save.rot4, main = "Howarth & Sinding-Larsen Stream Sediments
Open Data")
evaln(gx.rqpca.plot(sind.save.rot4, rowids = TRUE, main = "Howarth & Sinding-Larsen Stream Sediments
Open Data")
evaln(gx.rqpca.plot(sind.save.rot4, v1 = 3, v2 =4, rowids = TRUE, main = "Howarth & Sinding-Larsen Stream Sediments
Open Data")

## Clean-up

evaln(rm(sind.save))
evaln(rm(sind.save.rot4))
```

---

**gx.rotate**

*Function to Perform a Kaiser Varimax Rotation*

**Description**

Function to perform a Kaiser Varimax rotation on Principal Component (PCA) loadings and scores in an object saved from `gx.mva`, `gx.mva.closed`, `gx.robmva` or `gx.robmva.closed`. 

---

**Examples**

```r
## Make test data available
data(sind.mat2open)

## Generate gx.robmva.closed object
sind.save <- gx.robmva.closed(sind.mat2open)

## Display Mahalanobis distances
gx.md.plot(sind.save)

## Display default PCA results
gx.rqpca.screeplot(sind.save)
gx.rqpca.loadplot(sind.save)

## Display appropriately annotated results

gx.md.plot(sind.save, main = "Howarth & Sinding-Larsen
Stream Sediments, Opened Data", cex.main=0.8)
gx.rqpca.screeplot(sind.save, main = "Howarth & Sinding-Larsen Stream Sediments
Open Data")
gx.rqpca.plot(sind.save, rowids = TRUE, main = "Howarth & Sinding-Larsen Stream Sediments
Open Data")
sind.save$pvcontrib
gx.rqpca.plot(sind.save, v1 = 3, v2 =4, rowids = TRUE, main = "Howarth & Sinding-Larsen Stream Sediments
Open Data")

## Display Kaiser Varimax rotated (nrot = 4) results
sind.save.rot4 <- gx.rotate(sind.save, 4)
gx.rqpca.plot(sind.save.rot4, main = "Howarth & Sinding-Larsen Stream Sediments
Open Data")
gx.rqpca.plot(sind.save.rot4, rowids = TRUE, main = "Howarth & Sinding-Larsen Stream Sediments
Open Data")
gx.rqpca.plot(sind.save.rot4, v1 = 3, v2 =4, rowids = TRUE, main = "Howarth & Sinding-Larsen Stream Sediments
Open Data")

## Clean-up
rm(sind.save)
rm(sind.save.rot4)
```
Usage

gx.rotate(save, nrot = 2)

Arguments

save  
a saved object from the execution of function `gx.mva`, `gx.mva.closed`, `gx.robmva`, or `gx.robmva.closed`.

nrot  
the number of component loadings to be rotated, by default the first two components are rotated, nrot = 2.

Value

The value of nr is modified in, and the following are appended to, the object that was saved from `gx.mva`, `gx.mva.closed`, `gx.robmva`, or `gx.robmva.closed`:

nr  
modified to equal the number of components rotated.

vload  
the new loadings after Varimax rotation.

vscore  
the new scores after Varimax rotation.

vvcontrib  
the contribution of the rotated Varimax component to the total data variability.

pvvcontrib  
the contribution of the rotated Varimax component to the total data variability as a percentage.

cpvvcontrib  
the cumulative contribution of the rotated Varimax component to the total data variability as a percentage.

Author(s)

Robert G. Garrett

References


See Also

gx.mva, gx.mva.closed, gx.robmva, gx.robmva.closed, varimax, gx.rq pca.loadplot, gx.rq p ca.plot

Examples

```r
## Make test data available
data(sind.mat2open)

## Save PCA results and display biplots before and after Varimax rotation
sind.save <- gx.mva(clr(sind.mat2open))
gx.rq pca.plot(sind.save)
```
Function to Graphically Display PCA Loadings

Description

Function to graphically display PCA loadings computed by functions `gx.mva`, `gx.mva.closed`, `gx.robmva`, `gx.robmva.closed` or `gx.rotate`. The user may define the minimum absolute loading below which variables will not be graphically displayed, and modify the display title and text size as required.

Usage

```r
gx.rqpca.loadplot(save, main = "", crit = 0.3, cex = 0.8, cex.axis = 0.7, cex.main = 0.8)
```

Arguments

- **save**
  a saved object from any of functions `gx.mva`, `gx.mva.closed`, `gx.robmva` or `gx.robmva.closed`.

- **main**
  an alternate plot title from that generated automatically from information in the saved object, see Details below.

- **crit**
  the lower limit of the absolute value of a loading for a variable to be displayed, by default `crit = 0.3`.

- **cex**
  the text scale expansion factor for the variable names in the display, by default `cex = 0.8`, a 20% font size reduction.

- **cex.axis**
  the text scale expansion factor for the axis labels of the display, by default `cex.axis = 0.7`, a 30% font size reduction.

- **cex.main**
  the text scale expansion factor for the display title, by default `cex.main = 0.8`, a 20% font size reduction.
Details

If `main` is undefined the name of the matrix object supplied to the function is displayed, together with the value of `crit`. On the line below the name of the data matrix from which the PCA was derived is displayed. However, if an alternate plot title is preferred it may be defined, e.g., `main = "Plot Title Text"`. If no plot title is required set `main = " "`.

If the variable names are longer than three characters the display can easily become cluttered. In which case the user should redefine the variable names in the input matrix from which the PCA was derived using the `dimnames(matrix.name)[][2]` construct, and run the generating function again. Alternately, the variable names in the saved object may be changed directly via a redefinition of `save$matnames[[2]]`.

Author(s)

Robert G. Garrett, based on a script by Peter Filzmoser

References


See Also

gx.mva, gx.mva.closed, gx.robmva, gx.robmva.closed, gx.rotate

Examples

```r
## Make test data available
data(sind.mat2open)

## Estimate and display robust PCA loadings
sind.save <- gx.robmva.closed(sind.mat2open)  
gx.rqpca.loadplot(sind.save)

## Clean-up
rm(sind.save)
```

Function to Plot Principal Component Analysis Loadings and Scores

Description

Function to display the results of a Principal Components Analysis (PCA) from the saved object from `gx.mva, gx.mva.closed, gx.robmva, gx.robmva.closed` or `gx.rotate` as bi-plots. Various options for displaying loadings and scores are available, see Details below.
Usage

gx.rqpca.plot(save = TRUE, v1 = 1, v2 = 2, rplot = TRUE, qplot = TRUE, rowids = NULL, ifrot = TRUE, main = "", cex.lab = 0.9, cex.main = 0.9, rcex = 1, qcex = 0.8, rcol = 1, qcol = 1, ...)
Details

If `main` is undefined the name of the matrix object supplied to the function is displayed in the plot title. On the line below the name of the data matrix from which the PCA was derived is displayed. However, if an alternate plot title is preferred it may be defined, e.g., `main = "Plot Title Text"`. If no plot title is required set `main = " ".`

If the variable names are longer than three characters the display can easily become cluttered. In which case the user should redefine the variable names in the input matrix from which the PCA was derived using the `dimnames(matrix.name)[[2]]` construct, and run the generating function again. Alternately, the variable names in the saved object may be changed directly via a redefinition of `save$matnames[[2]]`. Information on the percentage of the variability explained by each component, and whether or not rotation has been undertaken, is recovered from the saved object and used to appropriately label the plot axes. Note that for non-robust models the percentage variability explained will be the same as the percentage variability explained by the corresponding eigenvalues.

The following describes the available plot option combinations, the first being the default:

- `rplot = TRUE & qplot = TRUE & rowids = NULL`, crosses (pch default) and variable names
- `rplot = TRUE & qplot = FALSE & rowids = NULL`, variable names only
- `rplot = FALSE & qplot = TRUE & rowids = NULL`, input matrix row numbers only
- `rplot = FALSE & qplot = TRUE & rowids = TRUE`, input matrix row identifiers
- `rplot = TRUE & qplot = TRUE & rowids = TRUE`, input matrix row numbers and variable names

Because functions `gx.mva, gx.mva.closed, gx.robmva or gx.robmva.closed` require a matrix as input the sample IDs that may be in a data frame may be lost. They may be re-inserted by copying `dimnames(...)[[1]]` from a data frame into the matrix. Alternately, to plot in the component score space with Sample IDs, the scores can be recovered from the saved object, e.g., `save$rqscore[, 1]` and `save$rqscore[, 2]`, and used as the x- and y-coordinates in function `xyplotNtags` with the sample IDs from the source data frame. Appropriate plot and axis titling can be displayed by setting the function arguments ‘by hand’.

Author(s)

Robert G. Garrett

References


See Also

`gx.mva, gx.mva.closed, gx.robmva, gx.robmva.closed, gx.rotate, xyplotNtags`

Examples

```r
## Make test data available
```
data(sind)
data(sind.mat2open)
attach(sind)

## Save PCA results and display bi-plots
sind.save <- gx.mva.closed(sind.mat2open)
gx.rqpca.plot(sind.save)
gx.rqpca.plot(sind.save, 
main = "Howarth & Sinding Larsen Stream Sediments\nclr transform",
pch = 4, rcol = 2, qcol = 4)
gx.rqpca.plot(sind.save, rplot = TRUE, qplot = FALSE, rowids = NULL)
gx.rqpca.plot(sind.save, rplot = FALSE, qplot = TRUE, rowids = NULL)
gx.rqpca.plot(sind.save, rplot = FALSE, qplot = TRUE, rowids = TRUE)
gx.rqpca.plot(sind.save, rplot = TRUE, qplot = TRUE, rowids = FALSE, 
rcol = 2, qcol = 4)

#
attach(sind)
xyplot.tags(sind.save$rqscore[, 1],sind.save$rqscore[, 2], ID, cex = 0.9)

## Clean-up and detach test data
rm(sind.save)
detach(sind)

---

gx.rqpca.print  

**Function to Display PCA Loadings and Scores**

**Description**

Function to display PCA matrices following computations by functions *gx.mva*, *gx.mva.closed*, *gx.robmva*, *gx.robmva.closed* or *gx.rotate*. The user may optionally display the loadings (default), the percentage contribution of the variables to the loadings, i.e. communalities (not default), and the scores on the PCs (default). Optionally the entire table of PC scores may be saved as a ‘.csv’ file for future use.

**Usage**

```r
gx.rqpca.print(save, ifload = TRUE, ifcntrb = FALSE, ifscore = TRUE, 
file = NULL)
```

**Arguments**

- `save`: a saved object from any of functions *gx.mva*, *gx.mva.closed*,
  *gx.robmva*, *gx.robmva.closed* or *gx.rotate*.
- `ifload`: if `ifload` = TRUE the PC loadings are displayed. The default is to display the
  PC loadings.
- `ifcntrb`: if `ifcntrb` = TRUE the percentage contribution of each variable (communality)
  to each PC is displayed. The default is not to display this table.
ifscore:  if ifscore = TRUE the scores on the PCs are displayed. The default is to display the PC loadings.

file:  the file name for saving the function output in the R working directory, see Details below.

Details

By default the PCA loadings and scores on the PCs are displayed on the current device. Optionally the percentage contribution, communality, of each variable to each PC may also be displayed. Additionally a table of cumulative percent contributions, communalities, is displayed to assist in deciding how many components to retain for rotation or further study. When the saved object from gx.rotate is the input object both the original and Varimax loadings and PC scores will be displayed by default. The last table displayed by the function may be saved as a ‘.csv’ file in the working directory. Note, the ‘.csv’ extension is appended in the function. See example below.

Value

The last displayed or saved table, table.rows, is returned and may be saved as an object if required.

Note

For large tables of scores all options may be set to FALSE to suppress table output to the display device, and the PC scores or rotated PC scores will be saved as a ‘.csv’ file as long as a text string is defined for file. If file is left undefined the function will fail with the message “object ‘table.rows’ not found”.

Author(s)

Robert G. Garrett

See Also

gx.mva, gx.mva.closed, gx.robmva, gx.robmva.closed, gx.rotate.

Examples

```r
## Make test data available
data(sind.mat2open)

## Estimate and display robust PCA loadings and scores
sind.save <- gx.robmva.closed(sind.mat2open)
gx.rqpca.print(sind.save, ifcntrb = TRUE)

## Save PCA scores for future use
gx.rqpca.print(sind.save, file = "sind.rob.pca.scores")

## Clean-up
rm(sind.save)
```
gx.rqpca.screeplot

Display a Scree Plot

Description

Function to display a scree plot arising from a Principal Components Analysis (PCA) from the saved object from `gx.mva`, `gx.mva.closed`, `gx.robmva` or `gx.robmva.closed`. In addition to the scree plot the cumulative variability explained is also displayed.

Usage

```r
gx.rqpca.screeplot(save, main = "", ...)```

Arguments

- `save`: a saved object from the execution of function `gx.mva`, `gx.mva.closed`, `gx.robmva` or `gx.robmva.closed`.
- `main`: an alternate plot title to that in the saved object, see Details below.
- `...`: further arguments to be passed to methods concerning the plot. For example, if some colour other than black is required for the plotting characters, specify `col = 2` to obtain red (see `display.lty` for the default colour palette). If it is required to make the plot title smaller, add `cex.main = 0.9` to reduce the font size by 10%.

Details

If `main` is undefined the name of the matrix object from which the PCA was derived is passed to the function via the saved object. Using the matrix name is the recommended procedure in the source functions as it helps to track the progression of the data analysis, acting as a record of the data source. However, at a presentation stage an alternate plot title may be preferred and can be defined in this function, e.g., `main = "Plot Title Text"`. If no plot title is required set `main = ""`.

Author(s)

Robert G. Garrett

See Also

`gx.mva`, `gx.mva.closed`, `gx.robmva`, `gx.robmva.closed`

Examples

```r
## Make test data available
data(sind.mat2open)

## Save PCA results and display scree plot
sind.save <- gx.mva(ilr(sind.mat2open))
gx.rqpca.screeplot(sind.save)`
The 'runs' test is used to infer whether two states, e.g., > and < some threshold are mutually independent along a traverse. In applied geochemical terms, it tests for pattern coherence. If the pattern of runs is not coherent at the scale of the sampling it will be difficult to identify any spatially consistent dispersion processes.

Usage

\[
gx\text{.runs}(n1, n2, u)
\]

Arguments

n1 the number of < threshold sites along a traverse.

n2 the number of > threshold sites along a traverse.

u the number of runs of > and < threshold sites along the traverse.

Note

Given a priori information on the location of a mineral occurrence, the \texttt{gx.hypergeom} function provides a far more insightful test. The 'runs' test is better suited for evaluating patterns due to lithological or environmental changes along a traverse when some ‘threshold’ can be selected that differentiates between two patterns.

Author(s)

Robert G. Garrett

References


See Also

\texttt{gx.hypergeom}
Examples

```r
## From Stanley (2003) Table 2

gx.runs(27, 4, 7)
gx.runs(25, 6, 7)
gx.runs(28, 3, 5)
```

---

**gx.scores**  
*Function to Compute Scores on the Basis of Threshold Estimates*

---

**Description**

Computes scores for a user selected group of variables based on the ratio of variable value to the  
variable threshold, i.e. the upper limit of background variation. The user must provide thresholds for  
the variables contributing to the scores. Optionally a set of relative weights may be provided that are  
applied to the scores. If above threshold values occur for a variable whose influence is indicative  
of a 'false' anomaly the relative weight for that variable may be set ‘-ve’, which will result in a  
reduction of the computed score. This function is a useful alternative to weighted sums when the  
variable data contains so many below DL values that summary statistics cannot be estimated. An  
object is created containing all the provided parameters and the scores for later reference and use.

**Usage**

```r
gx.scores(xx, tholds, rwts = NULL, setna = FALSE)
```

**Arguments**

- `xx`  
  name of the `n` by `p` matrix containing the data.

- `tholds`  
  a vector of the threshold estimates for the `p` variables.

- `rwts`  
  an optional vector of the relative weights for the `p` variables, negative weights  
  are permisable to indicate that high levels of the variable should have a negative  
  impact on the scores.

- `setna`  
  if it is required to set any `<0` scores to NAs then set `setna = TRUE`.

**Details**

If the data for only some of the variables available in an attached matrix or data frame are to be pro-  
cessed use the `cbind` construct. Thus, `temp.mat <- cbind(vname1, vname3, vname6, vname8)`,  
or the `cbind` may be used directly, see Example below. All computed scores with values less that 1  
are set to zero, optionally these may be replaced by NAs, to facilitate their removal from subsequent  
plots or maps.
Value

The following are returned as an object to be saved for further use:

- input: the name of the input data set
- tholds: the vector of thresholds used for the computations
- rwts: the vector of relative weights provided by the user
- scores: the computed scores

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Any rows in the data matrix with with NAs are removed prior to computing the weighted sums.

Author(s)

Robert G. Garrett

See Also

- `ltdl.fix.df`, `remove.na`

Examples

```r
## Make test data available
data(sind)
attach(sind)

## Compute scores - 1
sind.scores1 <- gx.scores(cbind(Cu, Zn, Cd), tholds = c(100, 200, 2))

## Compute scores - 2
sind.scores2 <- gx.scores(cbind(Cu, Zn, Cd, Fe), tholds = c(100, 200, 2, 2),
wts = c(1, 1, 1, -1), setna = TRUE)

## Detach test data
detach(sind)
```

Description

The function computes and displays a matrix of Robust ilr Stabilities (Filzmoser et al., 2010) and medians of log-ratios in the upper and lower triangles, respectively.
Usage

gx.sm(xx, ifwarn = TRUE)

Arguments

xx 
a matrix, or sub-matrix, of parts from a compositional data set.

ifwarn 
by default ifwarn = TRUE which generates a reminder/warning that when carrying out analyses of compositional data all data must be in the same measurement units. The message can be suppressed by setting ifwarn = FALSE.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df.

Any data vectors (rows) containing NAs are removed prior to computation.

This function is for used with closed compositional data sets, i.e. geochemical analyses. For the ‘classical’ Aitchison (1984, 1986) approach see gx.vm.

Author(s)

Robert G. Garrett

References


See Also

ltdl.fix.df.remove.na.ilr.stab.gx.vm

Examples

```r
## Make test data available
data(sind.mat2open)

## Compute Robust ilr Stabilities and log-ratio medians
gx.sm(sind.mat2open)
```
Function to Single Column Sort a Matrix or Data Frame

Description

Function to sort a matrix or data frame by the value in a column. On exit the function displays the sorted data. Any NAs in the sort column are sorted to beyond the greatest value. If the function is run as `temp <- gx.sort(x, ncol) the sorted data are not displayed, but retained in `temp for subsequent use or display.

Usage

```r
gx.sort(x, col = 1, reverse = FALSE)
```

Arguments

- `x`: the matrix or data frame to be sorted.
- `col`: a column number, the value of which will be used to sort the matrix or data frame.
- `reverse`: the default is to sort in ascending order of the value in column `col`. If a descending order sort is required, set `reverse = TRUE`.

Author(s)

Robert G. Garrett

See Also

- `gx.sort.df`

Examples

```r
## Make test data available
data(sind)
attach(sind)

## Sort data frame sind into ascending order on the value
## of column 4, Zn
gx.sort(sind, 4)

## Sort data frame sind into descending order on the value
## of column 4, Zn
gx.sort(sind, 4, reverse = TRUE)

## Sort only the geochemical data in data frame sind into
## descending order on the value of column 4, Zn. Note
## that sind[, -c(1:3)] moves the old column 4 to
## position 1
```
Function to Multi-Column Sort a Data Frame

Description

Function to sort a data frame on any combination of numerical values or factors in any combination of ascending or descending orders. If the function is run as temp <- gx.sort.df(formula, dfname) the sorted data are not displayed, but retained in temp for subsequent use or display.

Usage

gx.sort.df(formula, dfname)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>formula</td>
<td>a 'formula' defining the variables to be used in the sort and whether the sort for each is to be in ascending or descending order. The sort order is from left to right in the formula. See Details and Examples below.</td>
</tr>
<tr>
<td>dfname</td>
<td>the name of the data frame to be sorted.</td>
</tr>
</tbody>
</table>

Details

The sort is controlled by a text string in the form of a 'formula', so ~var1+var2 will sort in ascending order of var1, and then within equal values for var1 in ascending order of var2. A preceding + or - before a column name indicates a sort in ascending or descending order, respectively.

The function also works if formula and dfname are reversed in the function call.

Author(s)

Kevin Wright with some ideas from Andy Liaw

See Also

gx.sort
Examples

```r
## Make test data available
data(kola.c)
attach(kola.c)
names(kola.c)

## Create a small test data set for ID (1), COUNTRY (4),
## As (17), Co (21), Cu (23) and Ni (28)
test<-kola.c[1:25, c(1,4,17,21,23,28)]

## Sort test data into ascending order on the value of Ni
gx.sort.df(~Ni, test)
temp <- gx.sort.df(test, ~Ni)

temp

## Sort test data by Country and descending order of As
gx.sort.df(test, ~COUNTRY~As)

## Sort test data by Country and descending order of both
## As and Ni
gx.sort.df(test, ~COUNTRY~As~Ni)

## Clean-up and detach test data
rm(test)
rm(temp)
detach(kola.c)
```

---

gx.spearman

**Display Spearman Correlation Coefficients and their Significances**

**Description**

The function computes Spearman rank correlation coefficients and places them in the upper triangle of a printed matrix displayed on the current device, the probabilities that the coefficients are not due to chance (Ho: Coefficient = 0) are printed in the lower triangle. The diagonal is filled with NAs to visually split the two triangles.

**Usage**

```r
gx.spearman(xx, ifclr = FALSE, ifwarn = TRUE)
```

**Arguments**

- `xx` a matrix of numeric data.
- `ifclr` if `ifclr = TRUE` the data are Centred Log-Ratio transformed prior to the computation of the Pearson Coefficients. The default is no transformation.
- `ifwarn` by default `ifwarn = TRUE` which generates a reminder/warning that when carrying out a centred log-ratio transformation all the data must be in the same measurement units. The message can be suppressed by setting `ifwarn = FALSE`. 
Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df.

Any data vectors (rows) containing NAs are removed prior to computation.

This function is not recommended for use with closed compositional data sets, i.e. geochemical analyses, unless correlations are sought between a non-compositional variable and individual compositional variables. If it is used with compositional data, it is highly recommended that ifclr be set to TRUE to remove the effects of closure and display the ‘true’ inter-element variability. However, different groups of elements, subsets, of a data set will yield different inter-element correlations for the same pair of elements due to the nature of the clr transform. When carrying out a centred log-ratio transformation it is essential that the data are all in the same measurement units, and by default a reminder/warning is displayed if the data are centred log-ratio transformed, see ifwarn above.

For working with compositional data sets functions gx.vm and gx.sm are recommended. For visual displays see gx.pairs4parts and gx.plot2parts.

Author(s)

Robert G. Garrett

See Also

ltdl.fix.df, remove.na, clr

Examples

```r
## Make test data available
data(sind.mat2open)

## Compute Spearman correlation coefficients
gx.spearman(sind.mat2open)

## Note, unlike gx.pearson there is no example with a log
## transformation. The log transformation is monotonic
## and does not change the ranks

## Compute Spearman correlation coefficients following
## a centred log-ratio transformation
ngx.spearman(sind.mat2open, ifclr = TRUE)
```

---

**gx.stats**  
*Function to Compute and Display Summary Statistics*

Description

Function to compute summary statistics for a 'one-page' report and display in inset. Function may be used stand-alone, and is used as an 'engine' for the gx.summary.* series of functions.
Usage

gx.stats(xx, xlab = deparse(substitute(xx)), display = TRUE,
      iftell = TRUE)

Arguments

xx name of the variable to be processed.
xlab by default the character string for xx is used for the table title. An alternate title
      can be displayed with xlab = "text string", see Examples.
display if display = TRUE the summary statistics are displayed on the current device.
      If display = FALSE output is suppressed.
iftell by default the NA count is displayed by na.remove prior to the table of results
      from this function. When the function is used as a 'stats' engine the NA count
      display may be suppressed by the calling function when the NA count is to be
      displayed by that calling function.

Details

The summary statistics comprise the data minimum, maximum and percentile values, robust esti-
mates of standard deviation, the Median Absolute Deviation (MAD) and the Inter Quartile Standard
Deviation (IQSD), and the mean, variance, standard deviation (SD), coefficient of variation (CV%),
and the 95% confidence bounds on the median. When the minimum data value is > 0 summary
statistics are computed after a log10 data transformation and exported back to the calling function.

Value

stats the computed summary statistics to be used in function inset, and by
gx.summary.* functions. The list returned, stats, is a 32-element vector, see
      below:
[1:10] the minimum value, and the 1st, 2nd, 5th, 10th, 20th, 25th (Q1), 30th, 40th and
      50th (Q2) percentiles.
[11:19] the 60th, 70th, 75th (Q3), 80th 90th, 95th, 98th and 99th percentiles and the
      maximum value.
[20] the sample size, N.
[22] the Inter-Quartile Standard Deviation (IQSD).
[23] the data (sample) Mean.
[24] the data (sample) Variance.
[25] the data (sample) Standard Deviation (SD).
[26] the Coefficient of Variation as a percentage (CV%).
[27] the Lower 95% Confidence Limit on the Median.
[28] the Upper 95% Confidence Limit on the Median.
[29] the log10 transformed data (sample) Mean.
the log10 transformed data (sample) SD.
the log10 transformed data (sample) CV%.
If the minimum data value is <= 0, then stats[29:32] <- NA.

Note
Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df.
Any NAs in the data vector are removed prior to computation. Depending on the value of iftell, the NA count will be displayed, iftell = TRUE, or suppressed, iftell = FALSE.
The confidence bounds on the median are estimated via the binomial theorem, not by normal approximation.

Author(s)
Robert G. Garrett

See Also
ltdl.fix.df, remove.na

Examples

```r
## Make test data available
data(kola.o)
attach(kola.o)

## Generates an initial display
gx.stats(Cu)

## Provides a more appropriate labelled display
gx.stats(Cu, xlab = "Cu (mg/kg) in <2 mm 0-horizon soil")

## Detach test data
detach(kola.o)
```

---

**gx.subset**

*Extracts a Subset of Rows from a Data Frame*

**Description**

The function extracts a subset of rows, and columns if required, from a data frame and returns the subset as a new data frame based on the criterion provided by the user. Unused factor names are dropped.

**Usage**

```r
gx.subset(dfname, subset = TRUE)
```
Arguments

dfname    name of the data frame from which rows are to be extracted.
subset    the criterion for selecting the subset (rows).

Details

The subset criterion can be ‘complex’ and be a combination of conditions, see Examples below.

Value

data a data frame only containing the rows of the input data frame where the criterion is met.

Note

This function is based on a script shared by Bill Venables on S-News, October 10, 1997. As such it may pre-date the time that subset was added to the S-Plus library. It is simple to use and has been retained.

Author(s)

William N. Venables

See Also

subset

Examples

## Make test data available
data(kola.c)

## Make a subset of the data for Finland
finland.c <- gx.subset(kola.c, COUNTRY == "FIN")

## Make a subset of the data for rock type, LITHO 82 occurring
## in Russia. Note that both COUNTRY and LITHO are factor variables
russia.82 <- gx.subset(kola.c, COUNTRY == "RUS" & LITHO == 82)

## Make a subset of the data for Cu exceeding 50(ppm) in Norway
norway.cugt50 <- gx.subset(kola.c, COUNTRY == "NOR" & Cu >50)

## Make single element subsets, e.g. for use with function gx.cnpplts
## First locate the column in the data frame where the element of
## interest is stored using dimnames(kola.c)[[2]], we find that Be is
## the 19th column in the data frame

dimnames(kola.c)[[2]]
Norway <- gx.subset(kola.c,COUNTRY=="NOR")[,19]
Russia <- gx.subset(kola.c,COUNTRY=="RUS")[,19]
Finland <- gx.subset(kola.c,COUNTRY=="FIN")[,19]
gx.summary

Compiles a Table of Summary Statistics

Description
This function is a 'sub-engine' between the main summary statistics engine, 'gx.stats', and the
gx.summary.* display functions. Its 'sub-engine' function is to select the required results from the
gx.stats computations, and additionally compute 95% confidence bounds on means.

Usage
gx.summary(xx, log = log, iftell = iftell)

Arguments
xx 
name of the variable to be processed.

log 
if it will be required to display summary statistics following a log10 transformation of the data, set log = TRUE.

iftell 
passes the value of iftell for controlling the display of the NA count to function remove.na from the calling function.

Value
table 
a 15-element vector containing summary statistics, see below:
[1] the sample size, N.
[2] the number of NAs removed from the data passed for processing.
[3:7] the minimum value, Q1, Median, Q3 and maximum value.

The contents of elements [10:15] depend on the 'value' of log
[10] the data (sample) Mean.
[12] the Coefficient of Variation as a percentage (CV%).
[14] the Lower 95% Confidence Limit on the Mean.
the Upper 95% Confidence Limit on the Mean.

If \( \log = \text{TRUE} \), the results for the mean, \([13]\), and confidence limits, \([14:15]\), are backtransformed to the natural scale.

The returned table is rounded to 4 significant figures.

**Note**

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing `gx.summary.*` functions that call this function, see `ltdl.fix.df`.

Any NAs in the data vector will be removed prior to computation in function `gx.stats`. Depending on the value of `iftell`, the NA count will be displayed, `iftell = \text{TRUE}`; or suppressed, `iftell = \text{FALSE}`.

There are no examples for this function.

**Author(s)**

Robert G. Garrett

**See Also**

`gx.stats, ltdl.fix.df, remove.na`.

---

**gx.summary.groups**  
*Displays Summary Statistics for a Variable Grouped by a Factor*

**Description**

Displays the same concise one-line summary statistics report as `gx.summary1` but with the data grouped by the value of a factor variable. The table consists of a heading line and a line of summary statistics for each ‘group’, value of the factor variable. Optionally the data may be logarithmically (base 10) transformed.

**Usage**

```r
gx.summary.groups(group, x, xname = deparse(substitute(x)),
log = FALSE)
```

**Arguments**

- `group` : the name of the factor variable the data are to be grouped by.
- `x` : name of the variable to be processed.
- `xname` : by default the character string for `x` is used for the title. An alternate title can be displayed with `xname = \"text string\"`, see Examples.
- `log` : if the summary statistics are required following a log10 transformation, set `log = \text{TRUE}`.
Details
Setting log = TRUE results in a log transformation for the parametric statistical estimates. The maximum, minimum, quartiles and robust estimates of spread are estimated and reported in natural measurement units. Of the parametric statistics, the mean (the geometric mean) and 95% confidence are reported backtransformed into natural measurement units.

Note
Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df. Any NAs in the data vector will be removed prior to computation in function gx.stats. Display of the number of NAs found by function remove.na is suppressed in remove.na as the information is included in the display from this function.

Alternately, function framework.summary generates grouped summary statistics that are exported in a file format that can be directly imported into a spreadsheet, e.g., MS Excel, for inspection, or into other software, e.g., a Geographical Information System (GIS) where the spatial information concerning the ‘framework’ units is available, e.g., ecoclassification units.

For more extensive summary statistics displaying one variable at a time, see gx.summary2 using a construct like gx.summary2(var[factor == "value"]) or use function inset with a similar construct.

For summary graphical presentations see functions bwpots or tbplots.

Author(s)
Robert G. Garrett

See Also
gx.summary1, gx.summary, gx.stats, ltdl.fix.df, remove.na, gx.summary2

Examples
data(kola.c)
attach(kola.c)

## Generates an initial display
gx.summary.groups(COUNTRY, Cu)

## Provide a more informative display
gx.summary.groups(COUNTRY, Cu, xname = "Cu (mg/kg) in <2 mm 0-horizon soil")

## As above but with a log10 transformation to display
## the geometric mean, etc.
gx.summary.groups(COUNTRY, Cu, xname = "Cu (mg/kg) in <2 mm 0-horizon soil",
log = TRUE)

## Detach test data
detach(kola.c)
gx.summary.mat  Displays Summary Statistics for a Matrix or Data Frame

Description
Displays the same concise one-line summary statistics report as \texttt{gx.summary1} for two or more columns of a matrix or data frame. The table consists of a heading line and a line of summary statistics for each ‘variable’, column of the matrix or data frame. Optionally the data may be logarithmically (base 10) transformed.

Usage
\begin{verbatim}
gx.summary.mat(xmat, vars, banner = deparse(substitute(xmat)),
log = FALSE)
\end{verbatim}

Arguments
\begin{description}
\item[xmat] name of the matrix or data frame.
\item[vars] the indices, or names (see Example), of the columns of the matrix or data frame for the variables whose summary statistics are to be displayed.
\item[banner] by default the character string for \texttt{xmat}, the input matrix, is used for the title. An alternate title can be displayed with \texttt{banner = “text string”}, see Examples.
\item[log] if the summary statistics are required following a log10 transformation, set \texttt{log = TRUE}.
\end{description}

Details
Setting \texttt{log = TRUE} results in a log transformation for the parametric statistical estimates. The maximum, minimum, quartiles and robust estimates of spread are estimated and reported in natural measurement units. Of the parametric statistics, the mean (the geometric mean) and 95\% confidence are reported backtransformed into natural measurement units.

Note
Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see \texttt{ltdl.fix.df}. Any NAs in the data vector will be removed prior to computation in function \texttt{gx.stats}. Display of the number of NAs found by function \texttt{remove.na} is suppressed in \texttt{remove.na} as the information is included in the display from this function.

For a more extensive summary statistics display a variable at a time, see \texttt{gx.summary2}, and for a summary with graphical displays see \texttt{inset}.

For summary graphical presentations see functions \texttt{bwplots.by.var} or \texttt{tbplots.by.var}.

Author(s)
Robert G. Garrett
Display a one-line Summary Statistics Report

**Description**

Displays a concise one-line summary statistics report, below a heading line, consisting of sample size, number of NAs in the input vector; minimum, maximum and quartiles; robust estimates of the standard deviation (MAD and interquartile based measure); mean, standard deviation and coefficient of variation (%); and the standard error, and lower and upper 95% confidence limits on the mean. See Details for the results of setting `log = TRUE`. Optionally the data may be logarithmically (base 10) transformed.

**Usage**

```r
gx.summary1(xx, xname = deparse(substitute(xx)), log = FALSE)
```
Arguments

- `xx`: name of the variable to be processed.
- `xname`: by default the character string for `xx` is used for the title. An alternate title can be displayed with `xname = "text string"`, see Examples.
- `log`: if the summary statistics are required following a log10 transformation, set `log = TRUE`.

Details

Setting `log = TRUE` results in a log transformation for the parametric statistical estimates. The maximum, minimum, quartiles and robust estimates of spread are estimated and reported in natural measurement units. Of the parametric statistics, the mean (the geometric mean) and 95% confidence are reported backtransformed into natural measurement units. If all the results are required following a log10, or some other transformation, this can be achieved by executing the transformation in the call, e.g., `gx.summary1(log10(Cu))` or `gx.summary1(sqrt(Cu))`, and setting `log = FALSE`.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`. Any NAs in the data vector will be removed prior to computation in function `gx.stats`.

For a more extensive summary statistics display, see `gx.summary2`. For summary graphical displays see `shape` or `inset`.

Author(s)

Robert G. Garrett

See Also

`gx.summary1`, `gx.stats`, `ltdl.fix.df`, `remove.na`, `gx.summary2`

Examples

```r
# Make test data available
data(kola.o)
attach(kola.o)

# Generates an initial display
gx.summary1(Cu)

# Provide a more informative display
gx.summary1(Cu, xname = "Cu (mg/kg) in <2 mm Kola O-horizon soil")

# As above but with a log10 transformation to display
# the geometric mean, etc.
gx.summary1(Cu, xname = "Cu (mg/kg) in <2 mm Kola O-horizon soil", log = TRUE)

# Detach test data
detach(kola.o)
```
Display a ten-line Summary Statistics Report

Description
Displays a more extensive report than gx.summary1. The report includes sample size, number of NAs in the input vector; arithmetic mean and 95% confidence limits, standard deviation and CV%; geometric mean and 95% confidence limits, with standard deviation and CV% in log10 units; median and 95% confidence limits robust estimates of spread (MAD and interquartile based measure); and minimum, maximum, quartiles, and 2nd, 5th, 10th, 90th, 95th and 98th percentiles.

Usage
gx.summary2(xx, xname = deparse(substitute(xx)))

Arguments
- xx: name of the variable to be processed.
- xname: by default the character string for xx is used for the report title. An alternate title can be displayed with xname = "text string", see Examples.

Note
Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df. Any NAs in the data vector will be removed prior to computation in function gx.stats.
For a less extensive summary statistics display, see gx.summary1. For summary graphical displays see shape or inset.

Author(s)
Robert G. Garrett

See Also
gx.summary, gx.stats, ltdl.fix.df, remove.na, gx.summary1

Examples
```r
## Make test data available
data(kola.o)
attach(kola.o)

## Generates an initial display
gx.summary2(Cu)

## Provide a more informative display
gx.summary2(Cu, xname = "Cu (mg/kg) in <2 mm Kola 0-horizon soil")
```
## gx.triples.aov

**Carries out a 3-Level Staggered ANOVA and Estimates Variance Components**

### Description

Function to undertake an ANOVA for the unbalanced triplicates from a GSC NGR or Tri-National survey. The data must be in the following order for each triplicate: Analytical Duplicate, Field Duplicate for the Analytical Duplicate Split, other Field Duplicate. The results replicate those generated by the UANOVA (Garrett and Goss, 1980) computer program. Optionally the data may be logarithmically (base 10) transformed.

### Usage

```r
gx.triples.aov(x, xname = deparse(substitute(x)), log = FALSE, table = FALSE)
```

### Arguments

- **x**: a file of triplicate determinations, the order is critical, see Details below.
- **xname**: by default the character string for the data file name, `x`, is used for the table title. An alternate title can be displayed with `xname = "text string"`, see Examples.
- **log**: if a logarithmic transformation of the data is required to meet homogeneity of variance considerations (i.e. severe heteroscedasticity) set `log = TRUE`. This is also advisable if the range of the observations exceeds 1.5 orders of magnitude.
- **table**: set `table = TRUE` if the input data file is to be displayed. The default is no display.

### Details

As noted above, the order of the data is critical and must be as follows for each triplicate: Analytical Duplicate, Field Duplicate for the Analytical Duplicate Split, other Field Duplicate. The 'other Field Duplicate’ is equivalent to a regular regional-coverage sample, but is at a 'Field Duplicate’ site. Thus below, `x[i,1]` will contain the Analytical Duplicates, `x[i,2]` the Field Duplicates from which the Analytical Duplicates were split, and `x[i,3]` the other analytically unduplicated Field Duplicates. See Details in `triples.test1` for additional information.

### Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltal.fix.df`. Any NAs in the data must also be removed prior to running the `triples.aov` function. This requires care as the data must be in complete triplicate sets.
Author(s)

Robert G. Garrett

References


See Also

ltdl.fix.df, remove.na, triples.test1, gx.triples.fgx, triples.test2

Examples

```r
## Make test data available
data(triples.test1)
attach(triples.test1)

## Carry out unbalanced ANOVA
gx.triples.aov(Ba_ppm, xname =
"Ba (mg/kg - Aqua Regia digestion) in <2 mm unmilled C-horizon soil")

## Detach test data
detach(triples.test1)
```

Description

Function to execute a simple ANOVA to determine if the Field Duplicates are a valid subset of the regional coverage samples, and if the Field Duplicates pairs have ‘equivalent’ variability. Optionally the data may be logarithmically (base 10) transformed.

Usage

```r
gx.triples.fgx(x, RepStat, xname = deparse(substitute(x)),
log = FALSE)
```
Arguments

x a file of regional coverage and field duplicate data.

RepStat the Replicate Status code.

xname by default the character string for the data file name, x, is used for the table title. An alternate title can be displayed with xname = "text string", see Examples.

log if a logarithmic transformation of the data is required to meet homogeneity of variance considerations (i.e. severe heteroscedasticity) set log = TRUE. This is also advisable if the range of the observations exceeds 1.5 orders of magnitude.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df.

Any NAs in the data must also be removed prior to running the triples.fgx function. This requires care as the data must be in complete duplicate sets.

Author(s)

Robert G. Garrett

References


See Also

ltdl.fix.df, remove.na, triples.test1, gx.triples.fgx, triples.test2

Examples

## Make test data available
data(triples.test2)
attach(triples.test2)

## Carry out ANOVAs for equivalence of variances
gx.triples.fgx(Ba_ppm, RS, xname =
"Ba (mg/kg - Aqua Regia digestion) in <2 mm unmilled C-horizon soil")

## Detach test data
detach(triples.test2)
Display an Aitchison Variation Matrix for Compositional Data

Description

The function computes and displays an Aitchison Variation Matrix, with the variances and means of the log-ratios in the upper and lower triangles, respectively.

Usage

\texttt{gx.vm(xx, ifwarn = TRUE)}

Arguments

\texttt{xx} \quad \text{a matrix, or sub-matrix, of parts from a compositional data set.}
\texttt{ifwarn} \quad \text{by default } \texttt{ifwarn = TRUE} \text{ which generates a reminder/warning that when carrying out analyses of compositional data all data must be in the same measurement units. The message can be suppressed by setting } \texttt{ifwarn = FALSE}.\texttt{ifwarn = FALSE}.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see \texttt{ltdl.fix.df}.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data vector, must be removed prior to executing this function, see \texttt{ltdl.fix.df}.

Any data vectors (rows) containing \texttt{NAs} are removed prior to computation.

This function is for used with closed compositional data sets, i.e. geochemical analyses. For an alternate approach see \texttt{gx.sm}, where a robust ilr stability measure (Filzmoser et al., 2010) is used rather that the log-ratio variance, and the median of log-ratios is used rather than the mean.

Author(s)

Robert G. Garrett

References


See Also

ltdl.fix.df, remove.na, gx.sm

Examples

```r
### Make test data available
data(sind.mat2open)

### Compute Aitchison Variation Matrix
gx.vm(sind.mat2open)
```

### ilr

**Isometric Log-Ratio (ilr) transformation**

**Description**

Undertakes an isometric log-ratio transformation to remove the effects of closure in a data matrix.

**Usage**

```r
ilr(xx, ifclose = FALSE, ifwarn = TRUE)
```

**Arguments**

- `xx` a `n` by `p` matrix to be isometrically log-ratio transformed. It is essential that a single unit of measurement is used. Thus it may be required to convert, for example, determinations in percent to ppm (mg/kg) so that all measurements are in ppm prior to executing this function. Natural logarithms are used.
- `ifclose` if it is required to close a data set prior to transformation set `ifclose = TRUE`.
- `ifwarn` by default `ifwarn = TRUE` which generates a reminder/warning that when carrying out a centred log-ratio transformation all the data must be in the same measurement units. The message can be suppressed by setting `ifwarn = FALSE`.

**Details**

Most analytical chemical data for major, minor and trace elements are of a closed form, i.e. for a physical individual sample they sum to a constant, whether it be percent, ppm (mg/kg), or some other units. It does not matter that only some components contributing to the constant sum are present in the matrix, the data are closed. As a result, as some elements increase in concentration others must decrease, this leads to correlation measures and graphical presentations that do not reflect the true underlying relationships. However, isometrically transformed data are not suitable for univariate EDA inspection as the new synthetic variables bear a complex relationship to the original measurements. Other procedures for removing closure effects are additive log-ratios (alr) and centred log-ratios (clr).
Value

\[ x \] a \( (p-1) \) by \( p \) matrix of isometric log-ratio values. The names of the new \((p-1)\) synthetic variables, \(iso1\) through to \(isop\), where the \( p \) in \( isop \) equals \( p-1 \), are entered as column names in the matrix.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see \texttt{ltdl.fix.df}.

Any rows containing \texttt{NAs} in the data matrix are removed prior to undertaking the transformation.

The \texttt{ilr} transform is recommended for the calculation of Mahalanobis distances, a procedure which requires matrix inversion. When a Principal Component or Factor Analysis is required use of the \texttt{ilr} transform may be preferrable, see also the notes in \texttt{clr}. In that instance back transformation from the isometrically transformed variables to the original variables is required. Interested R users should refer to the papers by Filzmoser et al. (see below).

Author(s)

Peter Filzmoser and Karel Hron, with additions by Robert G. Garrett

References

Aitchison, J. and Egozcue, J.J., 2005. Compositional data analysis; where are we and where should we be heading. Mathematical Geology, 37(7):829-850.


See Also

\texttt{alr, clr, ltdl.fix.df, remove.na}

Examples

```r
## Make test data sind available
data(sind.mat2open)

## Undertake ilr transform
temp <- ilr(sind.mat2open)
temp

## Clean-up
```
ilr.stab

Compute the Robust ilr Stability for Two Parts of a Composition

Description

Function computes the Robust ilr Stability for two parts of a composition following the procedure in Filzmoser et al. (2010), see details below.

Usage

ilr.stab(xx1, xx2, ifwarn = T)

Arguments

xx1
   a column vector from a matrix or data frame of compositional data, xx1[1], ..., xx1[n].
xx2
   another column vector from the matrix or data frame of compositional data, xx2[1], ..., xx2[n]. xx1 and xx2 must be of identical length, n.
ifwarn
   by default ifwarn = TRUE which generates a reminder/warning that when carrying out analyses of compositional data all data must be in the same measurement units. The message can be suppressed by setting ifwarn = FALSE.

Details

The ilr transform of two parts of a composition is ilr.xy = 1/(sqrt(2)) * log(x1/x2). The Robust ilr Stability (Filzmoser et al., 2010) is computed from the MAD of the ilr.xy values. This is normalized into the (0,1) interval as exp(-ilr.MAD * ilr.MAD), following the procedure of Buccianti and Pawlowsky-Glahn (2005).

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df. Any data vectors (rows) containing NAs are removed prior to computation.

Author(s)

Robert G. Garrett

References


See Also

`ltdl.fix.df, remove.na`

Examples

```r
## Make test data available
data(sind)
attach(sind)

## Compute Robust ilr Stability
ilr.stab(Cu, Zn)

## Detach test data
detach(sind)
```

---

**inset**

**An EDA Graphical and Statistical Summary**

**Description**

Plots a three panel graphical distributional summary for a data set, comprising a histogram and a cumulative normal percentage probability (CPP) plot, together with a table of selected percentiles of the data and summary statistics between them. Optionally the EDA graphics may be plotted with base 10 logarithmic scaling.

**Usage**

```r
inset(xx, xlab = deparse(substitute(xx)), log = FALSE, xlim = NULL,
 nclass = NULL, colr = NULL, ifnright = TRUE, table.cex = 0.7, ...)
```

**Arguments**

- **xx**
  - name of the variable to be plotted.
- **xlab**
  - by default the character string for xx is used for the x-axis plot titles. An alternate title can be displayed with xlab = "text string", see Examples.
- **log**
  - if it is required to display the data with logarithmic (x-axis) scaling, set log = TRUE.
- **xlim**
  - default limits of the x-axis are determined in the function. However when used stand-alone the limits may be user-defined by setting xlim, see Note below.
- **nclass**
  - the default procedure for preparing the histogram is to use the Scott (1979) rule. This usually provides an informative histogram, other optional rules are nclass = "sturges" or nclass = "fd"; the later standing for Freedman-Diaconis (1981), a rule that is resistant to the presence of outliers in the data. See Venables and Ripley (2001) for details.
- **colr**
  - by default the histogram is infilled in grey, colr = 8. If no infill is required, set colr = 0. See function `display.lty` for the range of available colours.
controls where the sample size is plotted in the histogram display, by default this in the upper right corner of the plot. If the data distribution is such that the upper left corner would be preferable, set `ifnright = FALSE`.

table.cex on some display devices the table may be ‘cramped’ and the text lines overlap. If this is true `table.cex` can be decreased, the default is `table.cex = 0.7`, conversely it can be increased if the table appears ‘skinny’.

... further arguments to be passed to methods. For example, by default individual data points in the CPP plot are marked by a plus sign, `pch = 3`, if a cross or open circle is desired, then set `pch = 4` or `pch = 1`, respectively. See `display.marks` for all available symbols. Adding `ifqs = TRUE` results in horizontal and vertical dotted lines being plotted at the three central quartiles and their values, respectively, in the CPP plot.

Details

A histogram is displayed on the left, and a cumulative normal percentage probability plot on the right. Between the two is a table of simple summary statistics, computed by `gx.stats`, including minimum, maximum and percentile values, robust estimates of standard deviation, and the mean, standard deviation and coefficient of variation. The plots may be displayed with logarithmic axes, however, the summary statistics are not computed with a logarithmic transform.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Any NAs in the data vector are removed prior to displaying the plot.

If the default selection for `xlim` is inappropriate it can be set, e.g., `xlim = c(0, 200)` or `c(2, 200)`. If the defined limits lie within the observed data range a truncated plot will be displayed. If this occurs the number of data points omitted is displayed below the total number of observations.

The purpose of this function is to prepare publication quality graphics (.emf or .ps) files that can be included in reports or used as inset statistical summaries for maps. If a series of these are to be prepared the function `inset.exporter` can be used to advantage as it saves a graphics file as part of its procedure.

In some instances if the graphics window has been resized the last line(s) of the table may not be displayed, if so, resize the table until it is entirely visible. If the whole table is not visible it will not be saved properly to the graphics file in `inset.exporter`. Once as a complete graphics file the image may be resized in the receiving document.

For summary statistics tables to complement the graphical display see, `gx.stats`, `gx.summary1`, and `gx.summary2`.

In some R installations the generation of multi-panel displays and the use of function `eqscplot` from package MASS causes warning messages related to graphics parameters to be displayed on the current device. These may be suppressed by entering `options(warn = -1)` on the R command line, or that line may be included in a ‘first’ function prepared by the user that loads the ‘rgr’ package, etc.
inset.exporter

Author(s)

Robert G. Garrett

References


See Also

gx.hist.cnpllt, gx.stats, inset.exporter, ltdl.fix.df, remove.na

Examples

## Make test data available
data(kola.o)
attach(kola.o)

## Generates an initial display
inset(Cu)

## Provides a more appropriate display for publication
inset(Cu, xlab = "Cu (mg/kg) in <2 mm 0-horizon soil", log = TRUE)

## NOTE: The example statistics table may not display correctly

## Detach test data
detach(kola.o)

---

**inset.exporter**  
*Saves an EDA Graphical and Statistical Summary*

Description

Saves the output from function inset as a graphics file in the R working directory for use in report or map preparation. Optionally the EDA graphics may be plotted with base 10 logarithmic scaling.

Usage

inset.exporter(x, xlab = deparse(substitute(x)), log = FALSE, xlim = NULL, nclass = NULL, ifnright = TRUE, file = NULL, table.cex = 0.7, gtype = "emf", ...)
Arguments

x name of the variable to be plotted.

xlab a label for the x-axis. It is often desirable to replace the default x-axis label of the input variable name text string with a more informative label, e.g.,

\[
\text{xlab} = \text{"Cu (mg/kg) in <2 mm 0-horizon soil".}
\]

log if it is required to display the data with logarithmic (x-axis) scaling, set \(\text{log} = \text{TRUE}\).

xlim default limits of the x-axis are determined in the function. However when used stand-alone the limits may be user-defined by setting \(\text{xlim}\), see Note below.

nclass the default procedure for preparing the histogram is to use the Scott (1979) rule. This usually provides an informative histogram, other optional rules are \(\text{nclass} = \text{"sturges"} \) or \(\text{nclass} = \text{"fd"}\); the later standing for Freedman-Diaconis (1981), a rule that is resistant to the presence of outliers in the data. See \text{inset} or \text{gx.hist}.

ifnright controls where the sample size is plotted in the histogram display, by default this in the upper right corner of the plot. If the data distribution is such that the upper left corner would be preferable, set \(\text{ifnright} = \text{FALSE}\).

file the first part of the file name identifying the data source for saving the function output in the R working directory, see Details below.

table.cex on some display devices the table may be ‘cramped’ and the text lines overlap. If this is true \(\text{table.cex}\) can be decreased, the default is \(\text{table.cex} = 0.85\), conversely it can be increased if the table appears ‘skinny’.

gtype the format of the graphics file to be saved. By default \(\text{gtype} = \text{"emf"}\) for a Windows extended metafile. Other alternatives are \(\text{gtype} = \text{"jpg"}\) for a jpeg file, \(\text{gtype} = \text{"png"}\) for a portable network graphics file, \(\text{gtype} = \text{"ps"}\) for a postscript file, or \(\text{gtype} = \text{"pdf"}\) for a pdf file.

... further arguments to be passed to methods. For example, by default individual data points in the CPP plot are marked by a plus sign, \(\text{pch} = 3\), if a cross or open circle is desired, then set \(\text{pch} = 4\) or \(\text{pch} = 1\), respectively. See \text{display.marks} for all available symbols. Adding \(\text{ifqs} = \text{TRUE}\) results in horizontal and vertical dotted lines being plotted at the three central quartiles and their values, respectively, in the CPP plot.

Details

See \text{inset} for details concerning the \text{inset} parameters.

file contains the first part of the file name identifying the data source for the output file to be saved in the R working directory, see Note below. The function concatenates the working directory name with \(\text{file.deparse(substitute(x))}_\text{inset}\) as a character string for the file name. Subsequently the ‘value’ of \text{gtype} is appended as the file type and the file saved in the R working directory.

Note

To set the R working directory, if it has not already been set in a first function, use at the R command line, for example, \(\text{setwd("C:\R\Folder\")}\), where ‘n’ is some number, which will result in all saved output being placed in that folder.
Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Any NAs in the data vector are removed prior to displaying and saving the plot.

If the default selection for `xlim` is inappropriate it can be set, e.g., `xlim = c(0, 200)` or `c(2, 200)`, the latter being appropriate for a logarithmically scaled plot, i.e. `log = TRUE`. If the defined limits lie within the observed data range a truncated plot will be displayed. If this occurs the number of data points omitted is displayed below the total number of observations.

In some instances if the graphics window has been resized the last line(s) of the table may not be displayed, if so, resize the table until it is entirely visible. If the whole table is not visible it will not be saved properly to the graphics file in `inset.exporter`. Once as a complete graphics file the image may be resized in the receiving document.

In some R installations the generation of multi-panel displays and the use of function `eqscplot` from package MASS causes warning messages related to graphics parameters to be displayed on the current device. These may be suppressed by entering `options(warn = -1)` on the R command line, or that line may be included in a `first` function prepared by the user that loads the `rgr` package, etc.

**Author(s)**

Robert G. Garrett

**See Also**

`inset`, `ltdl.fix.df`

**Examples**

```r
## Make test data available
data(kola.o)
attach(kola.o)

## Usage is as follows
## inset.exporter(Cu, xlab = "Cu (mg/kg) in\n\n<2 mm O-horizon soil",
## log = TRUE, gtype = "wmf", file = "kola.o")

## Detach test data
detach(kola.o)
```

---

**kola.c**

*Kola Project C-horizon Soil Data*
Description

These data arise from an ecogeochemical survey undertaken by the Central Kola Expedition of Russia (CKE), the Geological Survey of Finland (GTK) and the Norwegian Geological Survey (NGU). In 1995 a variety of soil and biological materials were collected from almost 700 sites lying between the Arctic Circle and the Barents Sea, and Longitudes 35.5 and 40 East. This specific data set is for C-horizon soils found at 606 of the sites visited. The data consist of an integer identifier, Universal Transverse Mercator (m) eastings and northings coordinates, the country the site was located in as a 3 character string, the lithology of the underlying bedrock as an integer code, 36 chemical measurements (total or near-total geochemical analyses), and soil pH for the <2 mm fraction of the C-horizon soils. The data reflect the natural geochemical variations in the parent material of the overlying soils. Further details concerning the project, methods of sampling and analysis can be found in Reimann et al. (1998) and the numerous papers published by the co-authors in international scientific journals.

Usage

kola.c

Format

A data frame containing 44 observations for 617 sites.

Source

These data are a subset of the full Kola C-horizon data set available from: http://doi.pangaea.de/10.1594/PANGAEA.56227

However, it should be noted that the spatial coordinates are recorded as Latitudes and Longitudes in the full data set.

References

Universal Transverse Mercator (m) eastings and northings coordinates, 38 chemical measurements (total or near-total geochemical analyses), Loss on Ignition, soil pH and specific conductivity for the <2 mm fraction of the O-horizon (humus) soils. The data reflect both natural biogeochemical variations and the presence of heavy industry. Further details concerning the project, and methods of sampling and analysis can be found in Reimann et al. (1998) and the numerous papers published by the co-authors in international scientific journals.

Usage

kola.o

Format

A data frame containing 44 observations for 617 sites.

Source

These data are the same as in the R package 'mvoutlier'. However, note that the names of the spatial coordinates have been changed from XCOO and YCOO to UTME and UTMN, respectively, and COND (specific conductivity) to SC.

The full data set is available from: http://doi.pangaea.de/10.1594/PANGAEA.56279

However, it should be noted that this is a superset containing all geochemical analyses and the spatial coordinates are recorded as Latitudes and Longitudes in the full data set.

References


logit

Logit transformation

Description

Undertakes a logit transformation on a vector of proportions.

Usage

logit(pp)

Arguments

pp a vector of proportions in the range zero to one. The function may be used with a single proportion. Natural logarithms are used.
Details

Most analytical chemical data for major, minor and trace elements are of a closed form, i.e. for a sample they sum to a constant, whether it be percent, ppm (mg/kg), or some other units. It does not matter that only some components contributing to the constant sum are present in the matrix, the data are closed. As a result, as some elements increase in concentration others must decrease, this leads to statistics and graphical presentations that do not reflect the true underlying situation even in situations of univariate data analysis and display. The logit transformation is an appropriate transformation for univariate compositional data. However, for concentrations below 10% a logarithmic transform is sufficient. The inverse logit transform is the \texttt{expit}. Procedures for removing closure effects for multivariate data are additive log-ratios (\texttt{alr}), centred log-ratios (\texttt{clr}), and isometric log-ratios (\texttt{ilr}).

Value

\[ z \quad \text{a vector of the logit transformations of the proportions } p. \]

Note

If a value outside the range zero to one is encountered as a proportion the function displays an error message and halts.

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data matrix, must be removed prior to executing this function, see \texttt{ltdl.fix.df}.

If any NAs exist in the vector, pp, they are removed by function \texttt{remove.na} and the number removed is displayed.

Author(s)

Robert G. Garrett

References


See Also

\texttt{expit, alr, clr, ilr, ltdl.fix.df}

Examples

```r
## Generate test data
p <- c(0.1, 0.5, 0.9)

## Undertake and display logit transformations
z <- logit(p)
```

```r
z
```
# ltdl.fix

## Replace Negative Values Representing Less Than Detects in a Vector

### Description

Function to process a numeric vector to replace negative values representing less than detects (<DLs) with positive half that value. This permits processing of these effectively categorical data as real numbers and their display on logarithmically scaled axes. Some software packages replace blank fields that should be interpreted as NAs, i.e. no information, with zeros, the facility is provided to replace any zero values with NAs. In other instances data files have been built using an integer code, e.g., -9999, to indicate 'no data', i.e. the equivalent of NAs. The facility is provided to replace any so coded values with NAs. If required, all <values and may be replaced with NAs, e.g. when estimating analytical precision with `anova1` using only duplicate analyses with >DL values.

A report of the changes made is displayed on the current device.

For processing data matrices or data frames, see `ltdl.fix.df`.

### Usage

```
ltdl.fix(x, negs2na = FALSE, zero2na = FALSE, coded = NA)
```

### Arguments

- **x**
  - name of the vector to be processed.
- **negs2na**
  - to replace any -ve values with NAs, set `negs2na = TRUE`.
- **zero2na**
  - to replace any zero values with NAs, set `zero2na = TRUE`.
- **coded**
  - to replace any numeric coded values, e.g., -9999 with NAs, set `coded = -9999`.

### Value

A numeric vector identical to that input but where any negative values have been replaced by half their positive values, or optionally by NAs, and optionally any zero or numeric coded values have been replaced by NAs.

### Note

If data are being accessed through an ODBC link to a database, rather than from a data frame that can be processed by `ltdl.fix.df`, it may be important to run this function on the retrieved vector prior to any subsequent processing. The necessity for such vector processing can be ascertained using the range function, e.g., `range(na.omit(x))` or `range(x, na.rm = TRUE)`, where `x` is the variable name, to determine the presence of any negative values. The presence of any NAs in the vector will return NAs by `range` if they are not removed.
Great care needs to be taken when processing data where a large proportion of the data are less than detects (<value>). In such cases parametric statistics have limited value, and can be misleading. Records should be kept of variables containing <values, and the fixed replacement values changed in tables for reports to the appropriate <values. Thus, in tables of percentiles the <value should replace the fixed value computed from absolute(-value)/2. Various rules have been proposed as to how many less than detects treated in this way can be tolerated before means, variances, etc. become biased and of little value. Less than 5% in a large data set is usually tolerable, with greater than 10% concern increases, and with greater than 20% alternate procedures for processing the data should be sought, for example, the procedures outlined in Helsel (2005). Alternately replacement non-detect values may be imputed with R packages robCompositions (Hron et al., 2010; Templ et al., 2011) or zCompositions (Martin-Fernandez et al., 2012; Palarea-Albaladejo and Martin-Fernandez, 2013), specifically function lrem in the latter, both may be used with closed, constant-sum, data.

Author(s)

Robert G. Garrett

References


See Also

ltldl.fix.df

Examples

```r
## Replace any missing data coded as -9999 with NAs and any remaining
## negative values representing less than detects with Abs(value)/2
data(fix.test)
x.fixed <- ltlfix(fix.test[, 3], coded = -9999)
x.fixed

## As above, and replace any zero values with NAs
x.fixed <- ltlfix(fix.test[, 3], coded = -9999, zero2na = TRUE)
x.fixed
```
## ltl.div.df

Replace Negative Values Representing Less Than Detects in a Data Frame

### Description

Function to process a matrix or data frame to replace negative values representing less than detects (<value) with positive half that value. This permits processing of these effectively categorical data as real numbers and their display on logarithmically scaled axes. Some software packages replace blank fields that should be interpreted as NAs, i.e. no information, with zeros, the facility is provided to replace any zero values with NAs. In other instances data files have been built using an integer code, e.g., -9999, to indicate 'no data', i.e. the equivalent of NAs. The facility is provided to replace any so coded values with NAs. If required, all <values and may be replaced with NAs, e.g. when estimating analytical precision with anovaQ using only duplicate analyses with >DL values. Any factor variables in the input matrix or data frame are passed to the output matrix or data frame.

If a single vector is to be processed, use ltl.div

A report of the changes made is displayed on the current device.

### Usage

```r
ltl.div.df(x, negs2na = FALSE, zero2na = FALSE, coded = NA)
```
Arguments

x  
name of the matrix or data frame to be processed.

negs2na  
to replace any -ve values with NAs, set negs2na = TRUE.

zero2na  
to replace any zero values with NAs, set zero2na = TRUE.

coded  
to replace any numeric coded values, e.g., -9999 with NAs, set coded = -9999.

Value

A matrix or data frame identical to that input but where any negative values have been replaced by half their positive values, or NAs, and optionally any zero values or numeric coded values have been replaced by NAs.

Note

Great care needs to be taken when processing data where a large proportion of the data are less than detects (<value>). In such cases parametric statistics have limited value, and can be misleading. Records should be kept of variables containing <values, and the fixed replacement values changed in tables for reports to the appropriate <values. Thus, in tables of percentiles the <value should replace the fixed value computed from absolute(-value)/2. Various rules have been proposed as to how many less than detects treated in this way can be tolerated before means, variances, etc. become biased and of little value. Less than 5% in a large data set is usually tolerable, with greater than 10% concern increases, and with greater than 20% alternate procedures for processing the data should be sought. For example, the procedures outlined in Helsel (2005). Alternately replacement non-detect values may be imputed with R packages robCompositions (Hron et al., 2010; Templ et al., 2011) or zCompositions (Martin-Fernandez et al., 2012; Palarea-Albaladejo and Martin-Fernandez, 2013), specifically function lrEM in the latter, both may be used with closed, constant-sum, data.

Author(s)

Robert G. Garrett and David Lorenz

References


See Also

lttl.fix

Examples

## Replace any missing data coded as -9999 with NAs and any remaining
## negative values representing less than detects with Abs(value)/2
data(fix.test)
fix.test
fix.test.fixed <- ltdl.fix(df(fix.test, coded = -9999)
fix.test.fixed

## As above, and replace any zero values with NAs
fix.test.fixed <- ltdl.fix(df(fix.test, coded = -9999, zero2na = TRUE)
fix.test.fixed

## As above, but replace any negative values with NAs
fix.test.fixed <- ltdl.fix(df(fix.test, negs2na = TRUE, coded = -9999, zero2na = TRUE)
fix.test.fixed

## Clean-up
rm(fix.test.fixed)

map.eda7

Plot a Symbol Map of Data Based on the Tukey Boxplot

Description

Displays a simple map where the data are represented at their spatial locations by symbols using
Tukey boxplot-based symbology. Tukey boxplots divide data into 7 groups, the middle 50%, and
three lower and higher groupings, see Details below. The computation of the fences used to subdi-
vide the data may be carried out following a logarithmic transformation of the data. The colours of
the symbols may be optionally changed. Optionally a legend may be added to the map.

Usage

`map.eda7(xx, yy, zz, sfact = 1, logz = FALSE, xlab = "Easting",
ylab = "Northing", zlab = deparse(substitute(z)), main = "",
ifgrey = FALSE, symcolr = NULL, tol = 0.04, iflgn = FALSE,
title = deparse(substitute(zz)), ...)`

Arguments

xx name of the x-axis spatial coordinate, the eastings.

yy name of the y-axis spatial coordinate, the northings.

zz name of the variable to be plotted.

sfact controls the absolute size of the plotted symbols, by default sfact = 1. In-
creasing sfact results in larger symbols.
xlab  
a title for the x-axis, defaults to Easting.

ylab  
a title for the y-axis, defaults to Northing.

zlab  
by default, zlab = deparse(substitute(zz)), a map title is generated by appending the input variable name text string to "EDA Tukey Boxplot Map for ". Alternative titles may be generated, see Details below.

main  
an alternative map title, see Details below.

logz  
if it is required to undertake the Tukey Boxplot computations after a logarithmic data transform, set logz = TRUE.

ifgrey  
set ifgrey = TRUE if a grey-scale map is required, see Details below.

symcolr  
the default is a colour map and default colours are provided, deeper blues for lower values, green for the middle 50% of the data, and oranges and reds for higher values. A set of alternate symbol colours can be provided by defining symcolr, see Details below.

tol  
a parameter used to ensure the area included within the neatline around the map is larger than the distribution of the points so that the plotted symbols fall within the neatline. By default tol = 0.04, if more clearance is required increase the value of tol.

iflgnd  
the default is no legend. If a legend is required set iflgnd = TRUE, following the plotting of the data the cursor will be activated, locate that at the top left corner of the desired legend position and 'left button' on the pointing device.

title  
a short title for the legend, e.g., title = "Cu (mg/kg)". The default is the variable name.

...  
further arguments to be passed to methods. For example, if it is required to make the map title smaller, add cex.main = 0.9 to reduce the font size by 10%.

Details

Tukey boxplots divide data into 7 groups, the middle 50%, and three lower and higher groupings: within the whisker, near outliers and far outliers, respectively. Symbols for values below the first quartile (Q1) are plotted as increasingly larger circles, while symbols for values above the third quartile are plotted as increasingly larger squares, a '+' is used to plot the data falling in the middle 50%. For the higher groupings, the whisker contains values >Q3 and <=(Q3 + 1.5 * hw), where hw = (Q3 - Q1), the interquartile range; near outliers lie between (Q3 + 1.5 * hw) and (Q3 + 3 * hw); and far outliers have values >Q3 + 3 * hw. For the lower groupings the group boundaries, fences, fall similarly spaced below Q1. The computation of the fences used to subdivide the data may be carried out following a logarithmic transformation of the data, set logz = TRUE.

A summary table of the values of the symbol intervals, the number of values plotting as each symbol, and symbol shapes, sizes and colours is displayed on the current device.

If zlab and main are undefined a default a map title is generated by appending the input variable name text string to "EDA Tukey Boxplot-Based Map for ". If no map title is required set zlab = "", and if some user defined map title is required it should be defined in main, e.g. main = "Map Title Text".

If the grey-scale option is chosen the symbols are plotted 100% black for the far outliers, 85% black for the near outliers, 70% black for values within the whiskers, and 60% black for values falling within the middle 50% of the data.
The default colours, `symcolr = c(25, 22, 20, 13, 6, 4, 1)`, are selected from the rainbow(36) palette, and alternate colour schemes need to be selected from the same palette. See `display.rainbow` for the available colours. It is essential that 7 colours be provided, e.g., `symcolr = c(27, 24, 22, 12, 5, 3, 36)`, if exactly 7 are not provided the default colours will be displayed.

**Note**

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Any data vectors including NAs are removed prior to displaying the plot.

In some R installations the generation of multi-panel displays and the use of function eqscplot from package MASS causes warning messages related to graphics parameters to be displayed on the current device. These may be suppressed by entering `options(warn = -1)` on the R command line, or that line may be included in a ‘first’ function prepared by the user that loads the ‘rgr’ package, etc.

**Author(s)**

Robert G. Garrett

**See Also**

`display.rainbow`, `ltdl.fix.df`, `remove.na`

**Examples**

```r
## Make test data available
data(kola.o)
attach(kola.o)

## Plot a default symbol map
map.eda7(UTME, UTMN, Cu)

## Plot with logarithmically scaled boxplot fences and more
## appropriate axis labelling
map.eda7(UTME/1000, UTMN/1000, Cu, logz = TRUE,
  xlab = "Kola Project UTM Eastings (km)",
  ylab = "Kola Project UTM Northings (km)"
)

## Plot a grey-scale equivalent of the above map
map.eda7(UTME/1000, UTMN/1000, Cu, logz = TRUE, ifgrey = TRUE,
  xlab = "Kola Project UTM Eastings (km)",
  ylab = "Kola Project UTM Northings (km)"
)

## Plot the same map with an alternate colour scheme
map.eda7(UTME/1000, UTMN/1000, Cu, logz = TRUE,
  xlab = "Kola Project UTM Eastings (km)",
  ylab = "Kola Project UTM Northings (km)",
  symcolr = c(27, 24, 22, 12, 5, 3, 36))
```
## Detach test data

```
detach(kola.o)
```

---

### `map.eda8`  
*Plot a Symbol Map of Data Based on their Percentiles*

#### Description

Displays a simple map where the data are represented at their spatial locations by symbols indicating within which group defined by the data's 2nd, 5th, 25th, 50th, 75th, 95th and 98th percentiles plotted a data value falls. The colours of the symbols may be optionally changed. Optionally a legend (two options) may be added to the map.

#### Usage

```r
map.eda8(xx, yy, zz, sfact = 1, xlab = "Easting", ylab = "Northing",
        zlab = deparse(substitute(zz)), main = ",", ifgrey = FALSE,
        symcolr = NULL, tol = 0.04, iflgnd = FALSE, pctile = FALSE,
        title = deparse(substitute(zz)), 
...)
```

#### Arguments

- **xx**
  
  name of the x-axis spatial coordinate, the eastings.

- **yy**
  
  name of the y-axis spatial coordinate, the northings.

- **zz**
  
  name of the variable to be plotted.

- **sfact**
  
  controls the absolute size of the plotted symbols, by default `sfact = 1`. Increasing `sfact` results in larger symbols.

- **xlab**
  
  a title for the x-axis, defaults to `Easting`.

- **ylab**
  
  a title for the y-axis, defaults to `Northing`.

- **zlab**
  
  by default, `zlab = deparse(substitute(zz))`, a map title is generated by appending the input variable name text string to "EDA Percentile Based Map for " . Alternative titles may be generated, see Details below.

- **main**
  
  an alternative map title, see Details below.

- **ifgrey**
  
  set `ifgrey = TRUE` if a grey-scale map is required, see Details below.

- **symcolr**
  
  the default is a colour map and default colours are provided, deeper blues for lower values, green for the middle 50% of the data, and oranges and reds for higher values. A set of alternate symbol colours can be provided by defining `symcolr`, see Details below.

- **tol**
  
  a parameter used to ensure the area included within the neatline around the map is larger than the distribution of the points so that the plotted symbols fall within the neatline. By default `tol = 0.04`, if more clearance is required increase the value of `tol`. 

---

## Detach test data

```
detach(kola.o)
```
iflgnd

the default is no legend. If a legend is required set iflgnd = TRUE, following the plotting of the data the cursor will be activated, locate that at the top left corner of the desired legend position and ‘left button’ on the pointing device. There are two legends to choose from, see ptile below.

ptile

the default legend displays the range of values each symbol represents. Alternatively, the percentiles may be displayed rather than their values by setting ptile = TRUE.

title

a short title for the legend, e.g., title = "Cu (mg/kg)". The default is the variable name.

... further arguments to be passed to methods. For example, if it is required to make the map title smaller, add cex.main = 0.9 to reduce the font size by 10%.

Details

The selected percentiles, 2nd, 5th, 25th, 50th, 75th, 95th and 98th, divide the data into 8 groups. Values below the median are represented by increasingly larger deeper blue circles below the 25th percentile (Q1), and values above the 75th percentile (Q3) by increasingly larger orange and red squares. The mid 50% of the data are represented by green symbols, circles for the median (Q2) to Q1, and squares for the median (Q2) to Q3.

A summary table of the values of the symbol intervals, the number of values plotting as each symbol, and symbol shapes, sizes and colours is displayed on the current device.

If zlab and main are undefined a default a map title is generated by appending the input variable name text string to "EDA Percentile Based Map for " . If no map title is required set zlab = "", and if some user defined map title is required it should be defined in main, e.g. main = "Map Title Text".

If the grey-scale option is chosen the symbols are plotted 100% black for the far outliers, 85% black for the near outliers, 70% black for values within the whiskers, and 60% black for values falling within the middle 50% of the data.

The default colours, symcolr = c(25, 22, 20, 13, 13, 6, 4, 1), are selected from the rainbow(36) palette, and alternate colour schemes need to be selected from the same palette. See display.rainbow for the available colours. It is essential that 8 colours be provided, e.g., symcolr = c(27, 24, 22, 12, 12, 5, 3, 36), if exactly 8 are not provided the default colours will be displayed.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fdx.df.

Any data vectors including NAs are removed prior to displaying the plot.

In some R installations the generation of multi-panel displays and the use of function eqscplot from package MASS causes warning messages related to graphics parameters to be displayed on the current device. These may be suppressed by entering options(warn = -1) on the R command line, or that line may be included in a ‘first’ function prepared by the user that loads the ‘rgr’ package, etc.
Author(s)

Robert G. Garrett

See Also

display.rainbow, ltdl.fix.df, remove.na

Examples

```r
## Make test data available
data(kola.o)
attach(kola.o)

## Plot a default symbol map
map.eda8(UTME, UTMN, Cu)

## Plot a more appropriately labelled map
map.eda8(UTME/1000, UTMN/1000, Cu,
xlab = "Kola Project UTM Eastings (km)",
ylab = "Kola Project UTM Northings (km)"
)

## Plot a grey-scale equivalent of the above map
map.eda8(UTME/1000, UTMN/1000, Cu, ifgrey = TRUE,
xlab = "Kola Project UTM Eastings (km)",
ylab = "Kola Project UTM Northings (km)"
)

## Plot the same map with an alternate colour scheme
map.eda8(UTME/1000, UTMN/1000, Cu,
xlab = "Kola Project UTM Eastings (km)",
ylab = "Kola Project UTM Northings (km)",
symcolr = c(27, 24, 22, 12, 12, 5, 36))

## Detach test data
detach(kola.o)
```

map.tags

Plot a Map of Posted Values

Description

Displays a simple map where the data are represented by the ‘written’ values of the data at their spatial locations.

Usage

```
map.tags(xx, yy, tag, xlab = "Easting", ylab = "Northing",
taglab = deparse(substitute(tag)), main = "", tol = 0.04, ...)
```
Arguments

xx  name of the x-axis spatial coordinate, the eastings.

yy  name of the y-axis spatial coordinate, the northings.

tag  name of the variable to be plotted as a map.

xlab  a title for the x-axis, defaults to Easting.

ylab  a title for the y-axis, defaults to Northing.

taglab  by default, taglab = deparse(substitute(tag)), a map title is generated by appending the input variable name text string to "Map of Values for " . Alternative titles may be generated, see Details below.

main  an alternative map title, see Details below.

tol  a parameter used to ensure the area included within the neatline around the map is larger than the distribution of the points so that the plotted symbols fall within the neatline. By default tol = 0.04, if more clearance is required increase the value of tol.

...  further arguments to be passed to methods. For example, if smaller plotting characters are required, specify cex = 0.8; or if some colour other than black is required for the plotting characters, specify col = 2 to obtain red (see display.lty for the default colour palette). If it is required to make the map title smaller, add cex.main = 0.9 to reduce the font size by 10%.

Details

If taglab and main are undefined a default a map title is generated by appending the input variable name text string to "Map of Values for ". If no map title is required set taglab = "", or if an alternative to the variable name taglab is required it may be specified, taglab = "Alternative". If some user defined map title is required it should be defined in main, e.g. main = "Map Title Text" , in which instance taglab is ignored.

If a map of sample numbers, ‘IDs’, is required and they are not explicitly in the data frame, a map of data frame row numbers may be displayed by specifying dimnames(dfname)[[1]] as the value of tags.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df.

Any data vectors including NAs for spatial coordinates are removed prior to displaying the map, thus those ‘sites’ are not plotted. However, where coordinates are present any NAs in the variable to be plotted are replaced with a ‘+’ sign to indicate sites with ‘missing data’.

In some R installations the generation of multi-panel displays and the use of function eqscplot from package MASS causes warning messages related to graphics parameters to be displayed on the current device. These may be suppressed by entering options(warn = -1) on the R command line, or that line may be included in a ‘first’ function prepared by the user that loads the ‘rgr’ package, etc.
Author(s)

Robert G. Garrett

See Also

`ltdl.fix.df, remove.na, display.lty`

Examples

```r
## Make test data available
data(sind)
attach(sind)

## Plot a sample site number map
map.tags(E, N, ID)

## Plot a sample site number map with smaller numbers
## and a wider internal map margin
map.tags(E, N, ID, cex = 0.8, tol = 0.06)

## Plot the data frame row numbers rather than the sample
## numbers
map.tags(E, N, dimnames(sind)[[1]], cex = 0.8, tol = 0.06)

## Plot the values for Zn in red, providing enough internal
## map margin so the values do not overprint the map neat-line
map.tags(E, N, Zn, cex = 0.8, tol = 0.1, col = 2)

## Plot as above but with an informative title spread over
## two lines and with a slightly smaller font
map.tags(E, N, Zn, cex = 0.8, tol = 0.1, col = 2, main =
"Howarth & Sinding-Larsen\nStream Sediment Zn Data",
cex.main = 0.9)

## Detach test data
detach(sind)
```

Description

Displays a simple map where the data are represented by open circles whose diameters are proportional to the value of the data at their spatial locations. The rate of change of symbol diameter with value and the absolute size of the symbols are defined by the user. Optionally a legend may be displayed on the map.
map.z

Usage

map.z(xx, yy, zz, p = 0.5, sfact = 2.5, zmin = NA, zmax = NA,
       xlab = "Easting", ylab = "Northing",
       zlab = deparse(substitute(zz)), main = "", tol = 0.04,
       symcolr = 1, ifparams = FALSE, iflgnd = FALSE,
       title = deparse(substitute(zz)), ...) 

Arguments

xx     name of the x-axis spatial coordinate, the eastings.

yy     name of the y-axis spatial coordinate, the northings.

zz     name of the variable to be plotted as a map.

p      a parameter that controls the rate of change of symbol diameter with changing value. A default of \( p = 0.5 \) is provided. See Details below.

sfact  controls the absolute size of the plotted symbols, by default \( sfact = 2.5 \). Increasing sfact results in larger symbols.

zmin   a value below which all symbols will be plotted at the same minimum size. By default \( zmin = \) NA which results in the minimum value of the variable defining the minimum symbol size. See Details below.

zmax   a value above which all symbols will be plotted at the same maximum size. By default \( zmax = \) NA which results in the maximum value of the variable defining the maximum symbol size. See Details below.

xlab   a title for the x-axis, defaults to Easting.

ylab   a title for the y-axis, defaults to Northing.

zlab   by default, zlab = deparse(substitute(zz)), a map title is generated by appending the input variable name text string to "Proportional Symbol Map for " . Alternative titles may be provided, see Details below.

main   an alternative map title, see Details below.

tol    a parameter used to ensure the area included within the neatline around the map is larger than the distribution of the points so that the plotted symbols fall within the neatline. By default \( tol = 0.04 \), if more clearance is required increase the value of tol.

symcolr the colour of the symbols, the default is black, \( symcolr = 1 \). This may be changed if required, see display.lty for the default colour palette. For example, \( symcolr = 2 \) will cause the symbols to be plotted in red.

ifparams if ifparams = TRUE on completion of plotting and after the legend has been plotted, if requested, the cursor is activated, locate that at the top left corner of the desired text position and ‘left button’ on the pointing device. This text comprises three lines: the values of p to three significant figures and sfact; the maximum value of z to 3 significant figures and zmax; and the minimum value of z to 3 significant figures and zmin. The default is no text display.

iflgnd  the default is no legend. If a legend is required set iflgnd = TRUE, following the plotting of the data the cursor will be activated, locate that at the top left corner of the desired legend position and ‘left button’ on the pointing device. See Notes below.
title

A short title for the legend, e.g., title = "Zn (mg/kg)". The default is the variable name.

... further arguments to be passed to methods. For example, if smaller plotting characters are required for the legend, specify, for example, cex = 0.8; and if some other colour than black is required for the legend, specify, for example, col = 3, to obtain blue. Any colour change will also be reflected in the legend, if displayed. See display.lty for the default colour palette. If it is required to make the map title smaller, add cex.main = 0.9 to reduce the font size by 10%.

Details

The symbol diameter is computed as a function of the value $z$ to be plotted:

$$\text{diameter} = \text{dmin} + (\text{dmax} - \text{dmin}) \times \{(z - \text{zmin})/(\text{zmax} - \text{zmin})\}^p$$

where $\text{dmin}$ and $\text{dmax}$ are defined as 0.1 and 1 units, so the symbol diameters range over an order of magnitude (and symbol areas over two); $\text{zmin}$ and $\text{zmax}$ are the observed range of the data, or the range over which the user wants the diameters to be computed; and $p$ is a power defined by the user. The value of $(z - \text{zmin})/(\text{zmax} - \text{zmin})$ is the value of $z$ normalized, 0 - 1, to the range over which the symbol diameters are to be computed. After being raised to the power $p$, which will result in a number in the range 0 to 1, this value is multiplied by the permissible range of diameters and added to the minimum diameter. This results in a diameter between 0.1 and 1 units that is proportional to the value of $z$.

A $p$ value of 1 results in a linear rate of change. Values of $p$ less than unity lead to a rapid intial rate of change with increasing value of $z$ which is often suitable for displaying positively skewed data sets, see the example below. In contrast, values of $p$ greater than unity result in an initial slow rate of change with increasing value of $z$ which is often suitable for displaying negatively skewed data sets. Experimentation is usually necessary to obtain a satisfactory visual effect. See syms.pfunc for a graphic demonstrating the effect of varying the $p$ parameter.

The user may choose to transform the variable to be plotted prior to determining symbol size etc., e.g., $\log_{10}(zz)$, to generate a logarithmic rate of symbol size change. See Example below.

If $\text{zmin}$ or $\text{zmax}$ are defined this has the effect of setting a minimum or maximum value of $z$, respectively, beyond which changes in the value of $z$ do not result in changes in symbol diameter. This can be useful in limiting the effect of one, or a few, extreme outlier(s) while still plotting them, they simply plot at the minimum or maximum symbol size and are not involved in the calculation of the range of $z$ over which the symbol diameters vary. **Note:** If the variable $z$ includes a transform, e.g., $\log_{10}(z)$, the values of $\text{zmin}$ and/or $\text{zmax}$ must be in those transform units.

If $\text{zlab}$ and $\text{main}$ are undefined a default a map title is generated by appending the input variable name text string to "Proportional Symbol Map for " . If no map title is required set $\text{zlab} = "$", and if some user defined map title is required it should be defined in $\text{main}$, e.g. $\text{main} = "$Map Title Text"$.

**Note**

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df.

The legend consists of five proportional symbols and their corresponding $z$ values: $\text{zmin}$; the three quartiles; and $\text{zmax}$. If $\text{zmin}$ and $\text{zmax}$ have been user defined it is over their range that
the symbol sizes are computed and displayed. When defining \texttt{zmin} and/or \texttt{zmax} it is useful to set \texttt{ifparams = TRUE} as a reminder, whilst developing the required display.

Any NAs in the data vector are removed prior to displaying the plot.

In some R installations the generation of multi-panel displays and the use of function eqscplot from package MASS causes warning messages related to graphics parameters to be displayed on the current device. These may be suppressed by entering \texttt{options(warn=-1)} on the R command line, or that line may be included in a ‘first’ function prepared by the user that loads the ‘rgr’ package, etc.

\textbf{Author(s)}

Robert G. Garrett

\textbf{See Also}

\texttt{syms}, \texttt{syms.pfunc}, \texttt{ltdl.fix.df}, \texttt{remove.na}

\textbf{Examples}

```r
## Make test data available
data(kola.o)
attach(kola.o)

## Plot a default symbol map, p = 0.5 and sfact = 2.5
map.z(UTME, UTMN, Cu)

## Plot a map where the symbols are logarithmically scaled,
## and more appropriately labelled axes
map.z(UTME/1000, UTMN/1000, log10(Cu), p = 1,
    xlab = "Kola Project UTM Eastings (km)",
    ylab = "Kola Project UTM Northings (km)"
)

## Plot with differently scaled symbols and more appropriately
## labelled axes
map.z(UTME/1000, UTMN/1000, Cu, p = 0.3, sfact = 2.0,
    xlab = "Kola Project UTM Eastings (km)",
    ylab = "Kola Project UTM Northings (km)"
)

## Plot a map as above but where outliers above a value of 1000 are
## displayed with the same symbol
map.z(UTME/1000, UTMN/1000, Cu, p = 0.3, sfact = 2.0, zmax = 1000,
    xlab = "Kola Project UTM Eastings (km)",
    ylab = "Kola Project UTM Northings (km)"
)

## Detach test data
detach(kola.o)
```
Measurement Variability Test Data

Description

A data frame of magnetic susceptibility data used by Stanley (2003) to demonstrate the Thompson-Howarth procedure for estimating analytical variability (Thompson and Howarth, 1973 & 1978). They are used in the rgr package examples for the duplicate analysis ANOVA and Thompson-Howarth plot functions, `anova1` and `thplot1`, respectively. See also Garrett and Grunsky (2003).

Usage

`ms.data1`

Format

A data frame containing 2 measurements of magnetic susceptibility for each of 16 rock samples in 16 records.

Source

Stanley (2003), see below.

References


Measurement Variability Test Data

Description

A data frame of magnetic susceptibility data used by Stanley (2003) to demonstrate the Thompson-Howarth procedure for estimating analytical variability (Thompson and Howarth, 1973 & 1978). They are used in the rgr package examples for the duplicate analysis ANOVA and Thompson-Howarth plot functions, `anova2` and `thplot2`, respectively, with `ifalt = FALSE`. See also Garrett and Grunsky (2003).
Usage

ms.data2

Format

A data frame containing 2 measurements of magnetic susceptibility for each of 16 rock samples in 32 records. The measurements for the original analyses are in records 1 to 16, and the duplicate measurements are in records 17 to 32 in the same order.

Source

Stanley (2003), see below.

References


Description

A data frame of magnetic susceptibility data used by Stanley (2003) to demonstrate the Thompson-Howarth procedure for estimating analytical variability (Thompson and Howarth, 1973 & 1978). They are used in the rgr package examples for the duplicate analysis ANOVA and Thompson-Howarth plot functions, anovaR and thplotR, respectively. See also Garrett and Grunsky (2003).

Usage

ms.data3

Format

A data frame containing 2 measurements of magnetic susceptibility for each of 16 rock samples in 32 records. The measurements for the original and duplicate analyses alternate. So the first duplicate pair are in records 1 and 2, and the last in records 31 and 32.

Source

Stanley (2003), see below.
References


ogrady

Description

A data frame of major, minor and trace element data for the O’Grady pluton, NWT, (NTS map sheet 105I) from a regional lithogeochemical survey undertaken by the Geological Survey of Canada between 1969 and 1972 of Cretaceous-age granitoid plutons northeast of the Tintina Trench in the Yukon and adjoining NWT. Samples were collected in pairs from each site and a sub-sample ground to <100 mesh. Major and, some trace, element analyses were undertaken by direct reading optical spectroscopy (OES) either after a Li-Tetraborate (Li-T) fusion, or directly, other major elements were determined by AAS following HNO3 dissolution of the fusion product. Other trace-elements were determined by atomic absorption spectrophotometry after a HF-HClO4 digestion, or colorimetry (Col) after an alkaline fusion (AF). For Na, K, Fe, Mo and W detection limits (DLs) were 0.05, 0.1 and 0.1 %, and 0.5 and 2 mg/kg, respectively; <DL observations are represented by values of 0.02, 0.05, 0.05 % and 0.2 and 1 mg/kg, respectively.

Usage

data(ogrady)

Format

A data frame with 110 observations for the following 24 variables:

ID  a numeric vector, part of the unique GSC sample number.
E  UTM Eastings (m) for the sample site (UTM Zone 9).
N  UTM Northings (m) for the sample site (UTM Zone 9).
Lith  field name for the sampled lithology.
Si  silicon (%) in granitoid (Li-T OES).
Al  aluminium (%) in granitoid (Li-T OES).
Fe  iron (%) in granitoid (Li-T HNO3 AAS).
Mg  magnesium (%) in granitoid (Li-T OES).

Lithogeochemical Data Set from the O’Grady Pluton, NWT, 1970
Ca  calcium (%) in granitoid (Li-T OES).
Na  sodium (%) in granitoid (Li-T HNO3 AAS).
K  potassium (%) in granitoid (Li-T HNO3 AAS).
Ti  titanium (mg/kg) in granitoid (Li-T OES).
Mn  manganese (mg/kg) in granitoid (Li-T OES).
Ba  barium (mg/kg) in granitoid (Li-T OES).
Zn  zinc (mg/kg) in granitoid (HF-HClO4 AAS).
Cu  copper (mg/kg) in granitoid (HF-HClO4 AAS).
Pb  lead (mg/kg) in granitoid (HF-HClO4 AAS).
Mo  molybdenum (mg/kg) in granitoid (AF Col).
W  tungsten (mg/kg) in granitoid. (AF Col).
U  uranium (mg/kg) in granitoid (HF-HClO4 Fluorimetry).
Be  beryllium (mg/kg) in granitoid (OES).
V  vanadium (mg/kg) in granitoid (OES).
Sn  tin (mg/kg) in granitoid (OES).
Zr  zirconium (mg/kg) in granitoid (OES).

Source


References


ogrady.mat2open  Lithogeochemical Data Set from the O’Grady Pluton, NWT, 1970

Description

A matrix of major and minor element data (see below) for the O’Grady pluton, NWT, (NTS map sheet 105L), see ogrady for further details. Additionally, the UTM coordinates and lithology classifications have been removed from the ogrady data frame. This data set is provided so that users can investigate different approaches to the closure problem using the various functions in package ‘rgr’.
Usage

data(ogrady.mat2open)

Format

A matrix with 110 observations for the following 10 variables:

- **Si** silicon (mg/kg) in granitoid (Li-T OES).
- **Al** aluminium (mg/kg) in granitoid (Li-T OES).
- **Fe** iron (mg/kg) in granitoid (Li-T HNO3 AAS).
- **Mg** magnesium (mg/kg) in granitoid (Li-T OES).
- **Ca** calcium (mg/kg) in granitoid (Li-T OES).
- **Na** sodium (mg/kg) in granitoid (Li-T HNO3 AAS).
- **K** potassium (mg/kg) in granitoid (Li-T HNO3 AAS).
- **Ti** titanium (mg/kg) in granitoid (Li-T OES).
- **Mn** manganese (mg/kg) in granitoid (Li-T OES).
- **Ba** barium (mg/kg) in granitoid (Li-T OES).

Source


References


See Also

ogrady

---

**Computation of an Orthonormal Basis Matrix**

**Description**

Computes an orthonormal basis matrix to be used for the back-transformation of ilr-based data and statistics to clr-based data and statistics.
Usage
orthonorm(p)

Arguments
p

the dimension of the p-space, the number of original variables.

Value
V

the p by (p-1) orthonormal basis matrix.

Author(s)
Based on a function by Peter Filzmoser and Karel Hron

References

See Also
ilr, clr, gx.mva.closed, gx.robmva.closed, gx.md.gait.closed

Examples
## Make test data available
data(sind.mat2open)

## Compute and display clr transformed data
prmatrix(clr(sind.mat2open))

## Compute and display ilr transformed data
sind.ilr <- ilr(sind.mat2open)
prmatrix(sind.ilr)

## Compute and display orthonormal basis matrix
## sind.mat2open is a 25 by 6 matrix (data set)
V <- orthonorm(6)
prmatrix(V)

## Back-transform ilr transformed data to clr form and display
temp <- sind.ilr %*% t(V)
dimnames(temp)[[2]] <- dimnames(sind.mat2open)[[2]]
prmatrix(temp)

## Clean-up
rm(sind.ilr)
rm(V)
rm(temp)
remove.na  Remove and Count NAs

Description
Function to remove rows containing NAs from a data vector or matrix. Also counts the number of rows remaining, the number of rows deleted, and in the case of a matrix the number of columns. The results are returned in a list for subsequent processing in the calling function.

Usage
remove.na(xx, iftell = TRUE)

Arguments
xx  name of the vector or matrix to be processed.
iftell  if iftell = TRUE, the default, the number of removed records is displayed.

Details
This function is called by many of the procedures in the ‘rgr’ package. If one or more NAs are found the user is informed of how many. In general a data frame will have been cleared of any <values represented by negative values or zeros prior to executing the procedure calling this function, see ltdl.fix.df, or ltdl.fix if a single vector is being processed.

Value
x  a data vector or matrix containing the elements in the vector or rows of the matrix xx without NAs.
n  the length of x.
m  the number of columns in the matrix xx, if xx is a vector the value 1 is returned.
nna  the number of rows removed from xx.

Note
The iftell ‘switch’ is used to suppress the display of the NA count in some summary statistics tables as the information is included in the table.

Author(s)
Robert G. Garrett

See Also
ltdl.fix.df.where.na
Examples

## remove NAs
xx <- c(15, 39, 18, 16, NA, 53)
temp.x <- remove.na(xx)
x <- temp.x$x[1:temp.x$n]

## to recover the other values returned
n <- temp.x$n
m <- temp.x$m
nna <- temp.x$nna

## to remove NA replacing a -9999 in kola.o
data(kola.o)
kola.o.fixed <- ltdl.fix.df(kola.o, coded = -9999)
temp.x <- remove.na(kola.o.fixed$x)
x <- temp.x$x[1:temp.x$n]

## Clean-up
rm(xx)
rm(temp.x)
rm(x)
rm(n)
rm(m)
rm(nna)
rm(kola.o.fixed)

**Description**

Function to undertake a range transformation on a data matrix in order that each column is scaled zero-one between the minimum and maximum values.

**Usage**

rng(xx)

**Arguments**

xx  
a n by p matrix to be range transformed.

**Value**

x  
a n by p matrix of range-transformed values.
Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Any rows containing NAs in the data matrix are removed prior to undertaking the transformation.

A range transform may be appropriate for cluster analysis, including 2-d projection displays, applications to ensure all measured variables have equal weight.

Author(s)

Robert G. Garrett

See Also

`remove.na`

Examples

```r
## Make test data available
data(sind)
sind.mat <- as.matrix(sind[, -c(1:3)])

## Undertake range transform
temp <- rng(sind.mat)
temp

## Clean-up
rm(sind.mat)
rm(temp)
```

shape() An EDA Graphical Summary

Description

Plots a simple four panel graphical distributional summary for a data set, comprising a histogram, a horizontal Tukey boxplot or box-and-whisker plot (Garrett, 1988), an empirical cumulative distribution function (ECDF), and a cumulative normal percentage probability (CPP) plot. The plots in all four panels will have identical x-axis scaling. Optionally the EDA graphics may be plotted with logarithmic (base 10) scaling.

Usage

```r
shape(xx, xlab = deparse(substitute(xx)), log = FALSE, xlim = NULL, nclass = "Scott", ifbw = FALSE, wend = 0.05, ifnright = TRUE, colr = 8, cex = 0.8, ...)```

Arguments

xx  
name of the variable to be plotted.

xlab  
by default the character string for xx is used for the x-axis plot titles. An alternate
title can be displayed with xlab = "text string", see Examples.

log  
if it is required to display the data with logarithmic (x-axis) scaling, set log = TRUE.

xlim  
is determined by gxNhist and used to ensure all four panels in this function
have the same x-axis scaling. xlim may be defined, see Note below.

nclass  
the default procedure for preparing the histogram is to use the Scott (1979)
rule. This usually provides an informative histogram, other optional rules are
nclass = "sturges" or nclass = "fd"; the later standing for Freedman-
Diaconis (1981), a rule that is resistant to the presence of outliers in the data.
See Venables and Ripley (2001) for details.

ifbw  
the default is to plot a horizontal Tukey boxplot, if a box-and-whisker plot is
required set ifbw = TRUE.

wend  
if ifbw = TRUE the locations of the whisker-ends have to be defined. By default
these are at the 5th and 95th percentiles of the data, setting wend = 0.02 plots
the whisker ends at the 2nd and 98th percentiles.

colr  
by default the histogram and Tukey boxplot, or box-and-whisker plot, are in-
filled in grey, colr = 8. If no infill is required, set colr = 0. See function
displayNlty for the range of available colours.

ifnright  
controls where the sample size is plotted in the histogram display, by default this
in the upper right corner of the plot. If the data distribution is such that the upper
left corner would be preferable, set ifnright = FALSE.

cex  
by default the size of the text sample size, N, is set to 80%, i.e. cex = 0.8, and
may be changed if required.

...  
further arguments to be passed to methods. For example, the size of the axis
scale annotation can be changed by setting cexNaxis, the size of the axis titles
by setting cexNlab, and the size of the plot title by setting cexNmain. For exam-
ple, if it is required to make the plot title smaller, add cexNmain = 0.9 to reduce
the font size by 10%. By default individual data points in the ECDF and CPP
plots are marked by a plus sign, pch = 3, if a cross or open circle is desired,
then set pch = 4 or pch = 1, respectively. See displayNmarks for all available
symbols. Adding ifqs = TRUE results in horizontal and vertical dotted lines
being plotted at the three central quartiles and their values, respectively, in the
ECDF and CPP plots. By default the histogram and ‘box’ are infilled in grey,
colr = 8. If no infill is required, set colr = 0. See displayNlty for the range
of available colours.

Details

A histogram is displayed upper left, an ECDF is displayed below it (lower left). To the right of
the histogram a horizontal Tukey boxplot (default) or box-and-whisker plot (option) is displayed
(upper right). In the lower right quadrant a cumulative normal percentage probability (CPP) plot is
displayed. The x-axis scaling is identical in all four plots.
In a box-and-whisker plot there are two special cases. When \( w_{end} = 0 \) the whiskers extend to the observed minima and maxima that are not plotted with the plus symbol. When \( w_{end} = 0.25 \) no whiskers or the data minimum and maximum are plotted, only the median and box representing the span of the middle 50 percent of the data are displayed.

**Note**

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Any NAs in the data vector are removed prior to displaying the plots.

If the default selection for `xlim` is inappropriate it can be set, e.g., `xlim = c(0, 200)` or `c(2, 200)`, the latter being appropriate for a logarithmically scaled plot, i.e. `log = TRUE`. If the defined limits lie within the observed data range truncated plots will be displayed. If this occurs the number of data points omitted is displayed below the total number of observations in the various panels.

If it is desired to prepare a display of data falling within a defined part of the actual data range, then either a data subset can be prepared externally using the appropriate R syntax, or `xx` may be defined in the function call as, for example, `Cu[Cu < some.value]` which would remove the influence of one or more outliers having values greater than `some.value`. In this case the number of data values displayed will be the number that are `<some.value`.

In some R installations the generation of multi-panel displays and the use of function `eqscplot` from package MASS causes warning messages related to graphics parameters to be displayed on the current device. These may be suppressed by entering `options(warn = -1)` on the R command line, or that line may be included in a ‘first’ function prepared by the user that loads the ‘rgr’ package, etc.

For summary statistics displays to complement the graphics see, `gx.summary1`, `gx.summary2` and `inset`.

**Author(s)**

Robert G. Garrett

**References**


**See Also**

`gx.hist`, `bxplot`, `gx.ecdf`, `cnpplt`, `remove.na`, `display.lty`, `display.marks`, `ltdl.fix.df`, `inset`
sind

Examples

```r
## Make test data available
data(kola.o)
attach(kola.o)

## Generates an initial display to have a first look at the data and
## decide how best to proceed
shape(Cu)

## Provides a more appropriate initial display and indicates the
## quartiles
shape(Cu, xlab = "Cu (mg/kg) in <2 mm 0-horizon soil", log = TRUE,
ifqs = TRUE)

## Causes the Friedman-Diaconis rule to be used to select the number of
## histogram bins and changes the ECDF and CPP plotting symbols to a
## cross/x
shape(Cu, xlab = "Cu (mg/kg) in <2 mm 0-horizon soil", log = TRUE,
nclass = "fd", pch = 4)

## Replaces the Tukey boxplot with a box-and-whisker plot where the
## whiskers extend to the 10th and 90th percentiles and the minimum
## and maximum observed values are marked with a plus sign.
shape(Cu, xlab = "Cu (mg/kg) in <2 mm 0-horizon soil", log = TRUE,
ifbw = TRUE, wend = 0.1)

## Detach test data
detach(kola.o)
```

---

sind

**Sinding-Larsen Norwegian Stream Sediment Test Data Set**

Description

A data frame for a small subset of data from a regional geochemical stream sediment survey undertaken by the Norwegian Geological Survey.

Usage

```r
data(sind)
```

Format

A data frame with 25 observations on the following 9 variables:

- **ID** an arbitrary ID (numeric vector).
- **E** an Eastings coordinate.
- **N** a Northings coordinate.
Zn  zinc (mg/kg) in stream sediment.
Fe  iron (%) in stream sediment.
Mn  manganese (mg/kg) in stream sediment.
Cd  cadmium (mg/kg) in stream sediment.
Cu  copper (mg/kg) in stream sediment.
Pb  lead (mg/kg) in stream sediment.

Details

These data were used by Howarth and Sinding-Larsen (1983) to demonstrate the use of a number of multivariate statistical analysis techniques. Other authors, e.g., Howarth and Garrett (1986) and Garrett and Grunsky (2001) have also used the data set for demonstration purposes.

Source

Howarth and Sinding-Larson (1983), see below. Spatial coordinates added by digitizing Fig. 6-1 (op. cit.).

References


Data

sind.mat2open  Sinding-Larsen Norwegian Stream Sediment Test Data Set

Description

A matrix for a small subset of data from a regional geochemical stream sediment survey undertaken by the Norwegian Geological Survey. Similar to data set sind but the ID, Eastings and Northing columns have been removed, and the measurements are all in mg/kg.

Usage

data(sind.mat2open)
**Format**

A matrix with 25 observations on the following 6 variables:

- **Zn**  zinc (mg/kg) in stream sediment.
- **Fe**  iron (mg/kg) in stream sediment.
- **Mn**  manganese (mg/kg) in stream sediment.
- **Cd**  cadmium (mg/kg) in stream sediment.
- **Cu**  copper (mg/kg) in stream sediment.
- **Pb**  lead (mg/kg) in stream sediment.

**Details**

These data were used by Howarth and Sinding-Larsen (1983) to demonstrate the use of a number of multivariate statistical analysis techniques. Other authors, e.g., Howarth and Garrett (1986) and Garrett and Grunsky (2001) have also used the data set for demonstration purposes.

**Source**

Howarth and Sinding-Larson (1983), see below. Spatial coordinates added by digitizing Fig. 6-1 (op. cit.).

**References**


**See Also**

- **sind**
**Function to Compute the Diameters of Proportional Symbols**

**Description**

This function computes the diameters of the open circles to be plotted in a map or other display.

**Usage**

```plaintext
syms(z, zrange = c(NA, NA), p = 1)
```

**Arguments**

- `z`  
  name of the variable to be plotted for which diameters are to be computed.

- `zrange`  
  The minimum and maximum values of `z` to be used as the lower and upper limits, respectively, for the computed symbol diameters. By default the minimum and maximum values of the input data are used.

- `p`  
  a parameter that controls the rate of change of symbol diameter with changing value. A default of `p = 1` is provided that results in a linear rate of change. See Details below.

**Details**

The symbol diameter is computed as a function of the value `z` to be plotted:

\[
\text{diameter} = \text{dmin} + (\text{dmax} - \text{dmin}) \times \left(\frac{z - \text{zmin}}{\text{zmax} - \text{zmin}}\right)^p
\]

where `dmin` and `dmax` are defined as 0.1 and 1 units, so the symbol diameters range over an order of magnitude (and symbol areas over two); `zmin` and `zmax` are the observed range of the data, or the range over which the user wants the diameters to be computed; and `p` is a power defined by the user. The value of `(z - \text{zmin})/(\text{zmax} - \text{zmin})` is the value of `z` normalized, 0 - 1, to the range over which the symbol diameters are to be computed. After being raised to the power `p`, which will result in a number in the range 0 to 1, this value is multiplied by the permissible range of diameters and added to the minimum diameter. This results in a diameter between 0.1 and 1 units that is proportional to the value of `z`.

A `p` value of 1 results in a linear rate of change. Values of `p` less than unity lead to a rapid initial rate of change with increasing value of `z` which is often suitable for displaying negatively skewed data sets, see the example below. In contrast, values of `p` greater than unity result in an initial slow rate of change with increasing value of `z` which is often suitable for displaying positively skewed data sets. Experimentation is usually necessary to obtain a satisfactory visual effect. See `syms.pfunc` for a graphic demonstrating the effect of varying the `p` parameter.

If `zmin` or `zmax` are defined this has the effect of setting a minimum or maximum value of `z`, respectively, beyond which changes in the value of `z` do not result in changes in symbol diameter. This can be useful in limiting the effect of one or a few extreme outliers while still plotting them, they simply plot at the minimum or maximum symbol size and are not involved in the calculation of the range of `z` over which the diameter varies.
**Value**

zdiam the computed diameter of the symbol.

**Author(s)**

Robert G. Garrett

**See Also**

syms.pfunc

**Examples**

```r
## Make test data available
data(kola.o)attach(kola.o)

## Compute default symbol diameters
circle.diam <- syms(Cu, p = 0.3)
circle.diam

## Compute symbol diameters holding all symbols for values greater
## than 1000 to the same size
circle.diam <- syms(Cu, zrange = c(NA, 1000), p = 0.3)
circle.diam

## Clean-up and detach test data
rm(circle.diam)detach(kola.o)
```

**Description**

This function displays a plot demonstrating the effect of varying the value of p, for a range of p values from 0.2 to 5, on the 0 to 1 normalized values of a variable in order to compute corresponding circular symbol diameters.

**Usage**

`syms.pfunc()`

**Author(s)**

Robert G. Garrett
**tbplots**  

*Plot Vertical Tukey Boxplots*

**Description**

Plots a series of vertical Tukey boxplots where the individual boxplots represent the data subdivided by the value of some factor. Optionally the y-axis may be scaled logarithmically (base 10) and the values of the Tukey fences used to identify near and far outliers may also be optionally based on the logarithmically transformed data. A variety of other plot options are available, see Details and Note below.

**Usage**

```
tbplots(x, by, log = FALSE, logx = FALSE, notch = TRUE, xlab = "", ylab = deparse(substitute(x)), ylim = NULL, main = "", label = NULL, plot.order = NULL, xpos = NA, width, space = 0.25, las = 1, cex = 1, adj = 0.5, add = FALSE, ssll = 1, colr = 8, ...
```

**Arguments**

- **x**
  
  name of the variable to be plotted.

- **by**
  
  the name of the factor variable to be used to subdivide the data. See Details below for when by is undefined.

- **log**
  
  if it is required to display the data with logarithmic (y-axis) scaling, set `log = TRUE`.

- **logx**
  
  if the position of the Tukey boxplot fences are to be computed on the basis of the log transformed data set `logx = TRUE`. When `logx = TRUE` it is ensured that `log = TRUE`.

- **notch**
  
  determines if the boxplots are to be "notched" such that the notches indicate the 95% confidence intervals for the medians. The default is to notch the boxplets, to suppress the notches set `notch = FALSE`. See Details below.

- **xlab**
  
  a title for the x-axis, by default none is provided.

- **ylab**
  
  by default the character string for `x` is used for the y-axis title. An alternate title can be displayed with `xlab = "text string"`, see Examples.

- **ylim**
  
  only for `log = FALSE`, defines the limits of the y-axis if the default limits based on the range of the data are unsatisfactory. It can be used to ensure the y-axis scaling in multiple sets of boxplots are the same to facilitate visual comparison.

- **main**
  
  a main title may be added optionally above the display by setting `main`, e.g., `main = "Kola Project, 1995"`.

- **label**
  
  by default the character strings defining the factors are used to label the boxplots along the x-axis. Alternate labels can be provided with `label = c("Alt1", "Alt2", "Alt3")`. see Examples.

- **plot.order**
  
  provides an alternate order for the boxplots. Thus, `plot.order = c(2, 1, 3)` will plot the 2nd ordered factor in the 1st position, the 1st in the 2nd, and the 3rd in its 3rd ordered position, see Details and Examples below.
xpos  the locations along the x-axis for the individual vertical boxplots to be plotted. By default this is set to NA, which causes default equally spaced positions to be used, i.e. boxplot 1 plots at value 1 on the x-axis, boxplot 2 at value 2, etc., up to boxplot “n” at value “n”. See Details below for defining xpos.

width  the width of the boxes, by default this is set to the minimum distance between all adjacent boxplots times the value of space. With the default values of xpos this results in a minimum difference of 1, and with the default of space = 0.25 the width is computed as 0.25. To specify different widths for all boxplots use, for example, width = c(0.3). See Details below for changing individual boxplot widths.

space  the space between the individual boxplots, by default this is 0.25 x-axis units.

las  controls whether the x-axis labels are written parallel to the x-axis, the default las = 1, or are written down from the x-axis by setting las = 2. See also, Details below.

cex  controls the size of the font used for the factor labels plotted along the x-axis. By default this is 1, however, if the labels are long it is sometimes necessary to use a smaller font, for example cex = 0.8 results in a font 80% of normal size.

adj  controls the justification of the x-axis labels. By default they are centred, adj = 0.5, to left justify them if the labels are written downwards set adj = 0.

add  permits the user to plot additional boxplots into an existing display. It is recommended that this option is left as add = FALSE.

ssl1  determines the minimum data subset size for which a subset will be plotted. By default this is set to 1, which leads to only a circle with a median bar being plotted, as the subset size increases additional features of the boxplot are displayed. If ssl1 results in subset boxplots not being plotted, a gap is left and the factor label is still plotted on the x-axis.

colr  by default the boxes are infilled in grey, colr = 8. If no infill is required, set colr = 0. See display.1ty for the range of available colours.

...  further arguments to be passed to methods. For example, the size of the axis titles by setting cex.lab, and the size of the plot title by setting cex.main. For example, if it is required to make the plot title smaller, add cex.main = 0.9 to reduce the font size by 10%.

Details

There are two ways to execute this function. Firstly by defining x and by, and secondly by combining the two variables with the split function. See the first two examples below. The split function can be useful if the factors to use in the boxplot are to be generated at run-time, see the last example below. Note that when the split construct is used instead of by the whole split statement will be displayed as the default y-axis title. Also note that when using the by subsets are listed in the order that the factors are encountered in the data, but when using split the subsets are listed alphabetically. In either case they can be re-ordered using plot.order, see Examples.

The width option can be used to define different widths for the individual boxplots. For example, the widths could be scaled to be proportional to the subset population sizes as some function of the square root (const * sqrt(n)) or logarithm (const * log10(n)) of those sizes (n). The constant, const, would need to be chosen so that on average the width of the individual boxes would be
approximately 0.25, see Example below. It may be desirable for cosmetic purposes to adjust the positions of the boxes along the x-axis, this can be achieved by specifying `xpos`.

Long subset (factor) names can lead to display problems, changing the `las` parameter from its default of `las = 1` which plots subset labels parallel to the axis to `las = 2`, to plot perpendicular to the axis, can help. It may also help to use `label` and split the character string into two lines, e.g., by changing the string "Granodiorite" that was supplied to replace the coded factor variable `GRDR` to "Grano-\ndiorite". If this, or setting `las = 2`, causes a conflict with the x-axis title, if one is needed, the title can be moved down a line by using `xlab = \"Lithological Units\"`. In both cases the `\n` forces the following text to be placed on the next lower line.

If there are more than 7 labels (subsets) and no alternate labels are provided `las` is set to 2, otherwise some labels may fail to be displayed.

The notches in the boxplots indicate the 95% confidence intervals for the medians and can extend beyond the upper and lower limits of the boxes indicating the middle 50% of the data when subset population sizes are small. The confidence intervals are estimated using the binomial theorem. It can be argued that for small populations a normal approximation would be better. However, it was decided to remain with a non-parametric estimate despite the fact that the calculation of the Tukey fence values involves normality assumptions.

**Note**

This function is based on a script shared by Doug Nychka on S-News, April 28, 1992.

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Any NAs in the data vector are removed prior to preparing the boxplots.

For summary statistics displays to complement the graphics see `gx.summary.groups` or `framework.summary`.

**Author(s)**

Douglas W. Nychka and Robert G. Garrett

**See Also**

`cat2list`, `ltdl.fix.df`

**Examples**

```r
# Make test data kola.c available
data(kola.c)
attach(kola.c)

# Display a simple Tukey boxplot
tbplots(Cu, by = COUNTRY)
tbplots(spli(Cu,COUNTRY))

# Display a more appropriately labelled and scaled boxplot
tbplots(Cu, by = COUNTRY, log = TRUE, logx = TRUE, xlab = "Country",
ylab = "Ni (mg/kg) in <2 mm Kola C-horizon soil")

# Display a west-to-east re-ordered plot using the full country names
```
tbplots(split(Cu, COUNTRY), log = TRUE, logx = TRUE,
ylab = "Ni (mg/kg) in <2 mm Kola C-horizon soil",
label = c("Finland", "Norway", "Russia"),
plot.order = c(2, 1, 3))

## Detach test data kola.c
detach(kola.c)

## Make test data kola.o available, setting a -9999, indicating a
## missing pH measurement, to NA
data(kola.o)
kola.o.fixed <- ltlqfixdf(kola.o, coded = -9999)
attach(kola.o.fixed)

## Display relationship between pH in one pH unit intervals and Cu in
## 0-horizon (humus) soil
tbplots(split(Cu, trunc(pH+0.5)), log=TRUE, logx = TRUE,
xlab = "0-horizon soil pH to the nearest pH unit",
ylab = "Cu (mg/kg) in <2 mm Kola 0-horizon soil")

## As above, but demonstrating the use of variable box widths and the
## suppression of 95% confidence interval notches. The box widths are
## computed as (log10(n)+0.1)/5, the 0.1 is added as one subset has a
## population of 1. Note: paste is used in constructing xlab, below,
## as the label is long and overflows the text line length
## table(trunc(pH+0.5))
tbplots(split(Cu, trunc(pH+0.5)), log=TRUE, logx = TRUE, notch = FALSE,
xlab = paste("0-horizon soil pH to the nearest pH unit.",
"\bbox widths proportional to Log(subset_size)"),
ylab = "Cu (mg/kg) in <2 mm Kola 0-horizon soil",
width = c(0.26, 0.58, 0.24, 0.02))

## Detach test data kola.o.fixed
detach(kola.o.fixed)

---

tbplots.by.var  

### Plot Vertical Tukey Boxplots for Variables

**Description**

Plots a series of vertical Tukey boxplots where the individual boxplots represent the data subdivided by variables. Optionally the y-axis may be scaled logarithmically (base 10). A variety of other plot options are available, see Details and Note below.

**Usage**

```r
tbplots.by.var(xmat, log = FALSE, logx = FALSE, notch = FALSE,
xlab = "Measured Variables", ylab = "Reported Values",
main = "", label = NULL, plot.order = NULL, xpos = NA,
las = 1, cex = 1, adj = 0.5, colr = 8, ...)
```
Arguments

- **xmat**: the data matrix or data frame containing the data (variables).
- **log**: if it is required to display the data with logarithmic (y-axis) scaling, set `log = TRUE`.
- **logx**: if the positions of the Tukey boxplot fences are to be computed on the basis of log transformed data set `logx = TRUE`. When `logx = TRUE` it is ensured that `log = TRUE`.
- **notch**: determines if the boxplots are to be “notched” such that the notches indicate the 95% confidence intervals for the medians. The default is not to notch the boxplots, to have notches set `notch = TRUE`.
- **xlab**: a title for the x-axis, by default `xlab = "Measured Variables"`.
- **ylab**: a title for the y-axis, by default `ylab = "Reported Values"`.
- **main**: a main title may be added optionally above the display by setting `main`, e.g., `main = "Kola Project, 1995"`.
- **label**: by default the character strings defining the variables are used to label the boxplots along the x-axis. Alternate labels can be provided with `label = c("Alt1", "Alt2", "Alt3")`, see Examples.
- **plot.order**: provides an alternate order for the boxplots. By default the boxplot are plotted in alphabetical order of the factor variables. Thus, `plot.order = c(2, 1, 3)` will plot the 2nd alphabetically ordered factor in the 1st position, the 1st in the 2nd, and the 3rd in its alphabetically 3rd ordered position.
- **xpos**: the locations along the x-axis for the individual vertical boxplots to be plotted. By default this is set to `NA`, which causes default equally spaced positions to be used, i.e., boxplot 1 plots at value 1 on the x-axis, boxplot 2 at value 2, etc., up to boxplot “n” at value “n”. See Details below for defining `xpos`.
- **las**: controls whether the x-axis labels are written parallel to the x-axis, the default `las = 1`, or are written down from the x-axis by setting `las = 2`. See also, Details below.
- **cex**: controls the size of the font used for the factor labels plotted along the x-axis. By default this is 1, however, if the labels are long it is sometimes necessary to use a smaller font, for example `cex = 0.8` results in a font 80% of normal size.
- **adj**: controls the justification of the x-axis labels. By default they are centred, `adj = 0.5`, to left justify them if the labels are written downwards set `adj = 0`.
- **colr**: by default the boxes are infilled in grey, `colr = 8`. If no infill is required, set `colr = 0`. See `display.1ty` for the range of available colours.

... further arguments to be passed to methods. For example, the size of the axis titles by setting `cex.lab`, and the size of the plot title by setting `cex.main`. For example, if it is required to make the plot title smaller, add `cex.main = 0.9` to reduce the font size by 10%.

Details

There are two ways to provide data to this function. Firstly, if all the variables in a data frame are to be displayed, and there are no factor variables, the data frame name can be entered for `xmat`. 

...
However, if there are factor variables, or only a subset of the variables are to be displayed, the data are entered via the `cbind` construct, see Examples below.

Long variable names can lead to display problems, changing the `las` parameter from its default of `las = 1` which plots subset labels parallel to the axis to `las = 2`, to plot perpendicular to the axis, can help. It may also help to use `label` and split the character string into two lines, e.g., by changing the string "Specific Conductivity" that was supplied to replace the variable name `sc` to "Specific
Conductivity". If this, or setting `las = 2`, causes a conflict with the x-axis title, if one is needed, the title can be moved down a line by using `xlab = "\nPhysical soil properties"`. In both cases the `\n` forces the following text to be placed on the next lower line.

If there are more than 7 labels (variables) and no alternate labels are provided `las` is set to 2, otherwise some variable names may fail to be displayed.

The notches in the boxplots indicate the 95% confidence intervals for the medians and can extend beyond the upper and lower limits of the boxes indicating the middle 50% of the data when subset population sizes are small. The confidence intervals are estimated using the binomial theorem. It can be argued that for small populations a normal approximation would be better. However, it was decided to remain with a non-parametric estimate despite the fact that the calculation of the Tukey fence values involves normality assumptions.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.

Any NAs in the data vectors are removed prior to preparing the boxplots.

For a summary statistics display to complement the graphics see `gx.summary.mat`.

Author(s)

Robert G. Garrett

See Also

`tplots, var2fact, ltdl.fix.df`

Examples

```r
# Make test data kola.c available
data(kola.c)
attach(kola.c)

# Display a simple Tukey boxplot for measured variables
tplots.by.var(cbind(Co,Cu,Ni))

# Display a more appropriately labelled and scaled Tukey boxplot
tplots.by.var(cbind(Co,Cu,Ni), log = TRUE, logx = TRUE,
             ylab = "Contentrations (mg/kg) in <2 mm Kola C-horizon soil")

# Detach test data kola.c
detach(kola.c)
```
## Make test data ms.data available

data(ms.data)

## Display variables in a data frame, remembering to omit the sample IDs

tbplots.by.var(ms.data[, -1], log=TRUE, logx = TRUE)

---

**thplot1**

Display a Thompson-Howarth Plot of Duplicate Measurements

### Description

Function displays a Thompson-Howarth (1973 & 1978) plot for a set of duplicate measurements to visually inspect them as a part of the QA/QC process. By inputting a target precision the data may be visually checked to determine if they meet that criterion. The user is prompted for the location of the two legend items.

### Usage

`thplot1(x1, x2, xname = "", ifzero = 0.01, xlow = NA, xhih = NA, yhih = NA, rsd = 5, ptile = 95, main = "", ...)`

### Arguments

- `x1`: a column vector from a matrix or data frame, `x1[1], ..., x1[n]`
- `x2`: another column vector from a matrix or data frame, `x2[1], ..., x2[n]`. `x1`, `x2` must be of identical length, `n`, where `x2` is a duplicate measurement of `x1`.
- `xname`: by default the character string for `x1` is used for the title. An alternate title can be displayed with `xlab = "text string"`, see Examples.
- `ifzero`: as the Thompson-Howarth plot is log-scaled values of zero cannot be displayed, therefore the parameter `ifzero` has to be specified. A suitable choice is a value one order of magnitude lower than the value of the detection limit. A default value of `ifzero = 0.01` units is provided, corresponding to a detection limit of 0.1 units.
- `xlow`: if is desired to produce plots with consistent scaling this may be achieved by defining `xlow`, `xhih` and `yhih`, the `ylow` value is set equal to `ifzero`. Enter an appropriate value of `xlow` to ensure all data are displayed on all plots.
- `xhih`: enter an appropriate value of `xhih` to ensure all data are displayed on all plots.
- `yhih`: enter an appropriate value of `yhih` to ensure all data are displayed on all plots.
- `rsd`: to assist in QA/QC inspection a target precision may be defined as a RSD%, a default of `rsd = 5` is provided. See comments concerning RSD in Details below.
- `ptile`: defines the confidence interval for a line to be drawn on the plot above which only `100 - ptile%` of the points should plot if the defined target RSD is being met. A default of `ptile = 95` is provided. The function counts the number of points falling 'out of limits' and reports the probability that this number would have fallen 'out of limits' by chance alone.
main

... further arguments to be passed to methods. For example, the size of the axis scale annotation can be change by setting cex.axis, the size of the axis titles by setting cex.lab, and the size of the plot title by setting cex.main. For example, if it is required to make the plot title smaller, add cex.main = 0.9 to reduce the font size by 10%.

Details

This function expects the RSD% as a measure of repeatability (precision), which is more familiar to the current generation of applied geochemists, rather than the precision at the 2 Standard Deviation level. The necessary calculations to conform with the Thompson and Howarth procedure are made internally.

Duplicate pairs containing any NAs are omitted from the calculations.

If the data are as a single concatenated vector from a matrix or data frame as x[1], ..., x[n] followed by x[n+1], ..., x[2n], or alternated as x[1] and x[2] being a pair through to x[2*i+1] and x[2*i+2], for the i in 1:n duplicate pairs use function thplot2.

The user is prompted for the location of the two legend items added to the plot, the number of duplicate pairs, and whether or not the duplicates have met the RSD% criterion. In both instances the user is prompted for the location of left end of the text line, or the top left corner of the text block. The probability that the plotted data have met the RSD criterion is displayed in the text block; if the probability exceeds 0.9999 it is displayed as 0.9999.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df.

Duplicate pairs x1, x2 containing any NAs are omitted from the calculations.

This script was published by Garrett and Grunsky (2003)

Author(s)

Robert G. Garrett

References


See Also

thplot2, ltdl.fix.df, remove.na

Examples

```r
## NOTE: the examples below are commented out as thplot1 makes a
call to the locator function that fails when the examples are run
during package checking and building

## Make the Stanley (2003) test data available
data(ms.data1)
attach(ms.data1)

## Display the default plot
tthplot1(MS.1, MS.2, xname = "Magnetic Susceptibility",
## main = "Stanley (2003) Test Data")

## Display a Thompson-Howarth plot for a RSD of 7.5% and a draw the limit
## for a confidence interval of 90%
tthplot1(MS.1, MS.2, xname = "Magnetic Susceptibility", rsd = 7.5,
## ptile = 90, main = "Stanley (2003) Test Data")

## Detach test data
detach(ms.data1)
```

---

**thplot2**  
Display a Thompson-Howarth Plot of Duplicate Measurements, Alternate Input

**Description**

Function to prepare data stored in alternate forms from that expected by function thplot1 for its use. For further details see 'x' in Arguments below. The user is prompted for the location of the two legend items.

**Usage**

```r
thplot2(x, xname = deparse(substitute(x)), ifzero = 0.01,
xlow = NA, xhigh = NA, yhigh = NA, rsd = 5, ptile = 95, main = "",
ifalt = FALSE, ...)
```

**Arguments**

- `x`  
a column vector from a matrix or data frame, `x[1]`, ..., `x[2*n]`. The default is that the first `n` members of the vector are the first measurements and the second `n` members are the duplicate measurements. If the measurements alternate, i.e. duplicate pair 1 measurement 1 followed by measurement 2, etc., set `ifalt = TRUE`.  

by default the character string for x is used for the title. An alternate title can be displayed with `xlab = "text string"`, see Examples.

as the Thompson-Howarth plot is log-scaled values of zero cannot be displayed, therefore the parameter `ifzero` has to be specified. A suitable choice is a value one order of magnitude lower than the value of the detection limit. A default value of `ifzero = 0.01` units is provided, corresponding to a detection limit of 0.1 units.

if is desired to produce plots with consistent scaling this may be achieved by defining `xlow`, `xhih` and `yhih`, the `ylow`, the `ylow` value is set equal to `ifzero`. Enter an appropriate value of `xlow` to ensure all data are displayed on all plots.

enter an appropriate value of `xhih` to ensure all data are displayed on all plots.

enter an appropriate value of `yhih` to ensure all data are displayed on all plots.

to assist in QA/QC inspection a target precision may be defined as a RSD%, a default of `rsd = 5` is provided. See comments concerning RSD in details below.

defines the confidence interval for a line to be drawn on the plot above which only 100 - `ptile`% of the points should plot if the defined target RSD is being met. A default of `ptile = 95` is provided. The function counts the number of points falling `out of limits` and reports the probability that this number would have fallen `out of limits` by chance alone.

a title may be added optionally above the display, see Example.

set `ifalt = TRUE` to accommodate alternating sets of paired observations.

further arguments to be passed to methods. For example, the size of the axis scale annotation can be change by setting `cex.axis`, the size of the axis titles by setting `cex.lab`, and the size of the plot title by setting `cex.main`. For example, if it is required to make the plot title smaller, add `cex.main = 0.9` to reduce the font size by 10%.

This function expects the RSD% as a measure of repeatability (precision), which is more familiar to the current generation of applied geochemists, rather than the precision at the 2 Standard Deviation level. The necessary calculations to conform with the Thompson and Howarth procedure are made internally.

For further details see `thplot1`.

Duplicate pairs containing any NAs are omitted from the calculations.

If the data are as n duplicate pairs, x1 and x2, use function `thplot1`.

The user is prompted for the location of the two legend items added to the plot, the number of duplicate pairs, and whether or not the duplicates have met the RSD% criterion. In both instances the user is prompted for the location of the left end of the text line, or the top left corner of the text block.

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`.
Author(s)

Robert G. Garrett

See Also

thplot, ltdl.fix.df

Examples

## NOTE: the examples below are commented out as thplot1 makes a
## call to the locator function that fails when the examples are run
## during package checking and building

## Make test data available
##data(ms.data2)
##attach(ms.data2)

## Display the default plot
##thplot2(MS, xname = "Magnetic Susceptibility",
## main = "Stanley (2003) Test Data")

## Detach test data
##detach(ms.data2)

## Make test data available
##data(ms.data3)
##attach(ms.data3)

## Display a Thompson-Howarth plot for a RSD of 7.5% and draw
## the limit for a confidence interval of 90%
##thplot2(MS, xname = "Magnetic Susceptibility", rsd = 7.5, ptile = 90,

## Detach test data
##detach(ms.data3)

---

triples.test1 North American Geochemical Soil Landscapes Project QA/QC data

Description

A data frame of QA/QC triplicates samples from the Maritimes 2007 North Americac Soil Geochemical Landscapes survey. The analyses are for <2 mm unmilled C-horizon soil samples, and the determinations were made by ICP-OES or -MS following an Aqua Regia digestion.

Usage

data(triples.test1)
Format

A data frame with 27 observations for the following 3 variables:

ID the NAmSGLs unique site identifier.
RS the Replicate Status code.
Ba_ppm the barium determinations, mg/kg.

Details

The Replicate Status code indicates the ‘position’ of the geochemical sample in the QA/QC structure. RS = 8 indicates analytical duplicate, RS = 2 indicates the field duplicate, and RS = 1 indicates a routine regional coverage site that was ‘duplicated’. All other routine regional coverage sites are coded RS = 0. The analytical duplicate may be split from either of the two field sites, this information being in the project database. Thus the ‘triples’ may occur in the sequence ‘8, 2, 1’ or ‘8, 1, 2’. For gx.triples.aov to estimate the variance components correctly the geochemical samples must occur in the file in correct sequence.

Source

The Geological Survey of Canada, see Open File 6433, from which this QA/QC subset for barium (Ba) was extracted.

References


Description

A data frame of regional and field duplicate samples from the Maritimes 2007 North American Soil Geochemical Landscapes survey. The analyses are for <2 mm unmilled C-horizon soil samples, and the determinations were made by ICP-OES or -MS following an Aqua Regia digestion.

Usage

data(triples.test2)

Format

A data frame with 186 observations for the following 3 variables:

ID the NAmSGLs unique site identifier.
RS the Replicate Status code.
Ba_ppm the barium determinations, mg/kg
Details

The Replicate Status code indicates the ‘position’ of the geochemical sample in the QA/QC structure. RS = 2 indicates the field duplicate, and RS = 1 indicates a routine regional coverage site that was ‘duplicated’. All other routine regional coverage sites are coded RS = 0.

Source

The Geological Survey of Canada, see Open File 6433, from which this data subset for barium (Ba) was extracted.

References


---

var2fact

Rearranges Data for Variables as Factors

Description

Rearranges data from a matrix or data frame into a matrix where data are tagged by their variables names as factors. Used to concatenate data for display with functions tbplots.by.var and bwplots.by.var.

Usage

var2fact(xmat)

Arguments

xmat name of the n by p data matrix or data frame to be processed.

Details

If the data for only some of the variables available in an attached matrix or data frame are to be processed use the cbind construct. Thus, temp.mat <- cbind(vname1, vname3, vname6, vname8).

Value

xx a n * p by 2 matrix where each of the n * p rows contains a value that is paired with its variable name as a factor, see Note below.

Note

The p variables for n cases results in a n * p by 2 matrix, where [1:n, 1] contains the variable name for value[1] and [1:n, 2] contains the values for the n rows in the first column of xmat. Then rows [n+1:2n, 1] contain the variable name for value[2] and [n+1:2n, 2] contain the values for n rows in the second column, and so on.
Author(s)

Robert G. Garrett

Examples

```r
## Display, convert data frame and display the result
data(ms.data1)
ms.data1
temp <- var2fact(ms.data1)
temp

## Clean-up
rmo(temp)
```

where.na  Identify Vector Elements or Data Frame/Matrix Rows with NAs

Description

Function to display the positions of elements in a vector containing NAs, or the numbers of rows in a data frame or matrix containing one or more NAs. The function can also be used to remove NAs.

Usage

```r
where.na(x)
```

Arguments

- `x` name of the vector or matrix/data frame to be processed.

Value

- `whichna` a vector containing the indices of the positions in `x` containing NAs.

Note

This function is based on the S-Plus function `which.na` and is useful in finding the location of NAs in a data set. While `remove.na` removes NAs it does not identify their positions. A vector is returned that can be used to remove NAs, see example below.

Remember, a matrix is also a vector with the columns occurring sequentially.

Author(s)

S-Plus team and Robert G. Garrett

See Also

`remove.na`
Examples

```r
## Identify rows with NAs
xx <- c(15, 39, 18, 16, NA, 53)
where.na(xx)

## To use where.na to remove NAs, method 1
xx
temp <- where.na(xx)
temp
xxx <- xx[-temp]
xxx

## To use where.na to remove NAs, method 2
xx
xxx <- xx[-where.na(xx)]
xxx

## Clean-up
rm(xx)
rm(xxx)
rm(temp)
```

---

### wtd.sums

**Function to Compute Weighted Sums**

**Description**

Computes weighted sums for a user selected group of variables (Garrett et al., 1980; Garrett and Grunsky, 2001). The user must provide the relative importances of the variables contributing to the weighted sums. By default the median and MAD are estimated as measures of location and spread for the data. These may be replaced with alternate estimates if the user wishes, see Details below. An object is created containing all the estimated parameters and the weighted sums for later reference and use.

**Usage**

```r
wtd.sums(xx, ri, xloc = NULL, xspread = NULL)
```

**Arguments**

- **xx**
  - name of the \( n \times p \) matrix containing the data.

- **ri**
  - a vector of the relative weights for the \( p \) variables, negative weights are permissible to indicate that high levels of the variable should have a negative impact on the weighted sums.

- **xloc**
  - the default procedure is to use the computed medians of the input variables. Alternately, a vector of \( p \) estimates of location may be provided.

- **xspread**
  - the default procedure is to use the computed MADs of the input variables. Alternately, a vector of \( p \) estimates of spread may be provided.
Details

If the data for only some of the variables available in an attached matrix or data frame are to be processed use the cbind construct. Thus, `temp.mat <- cbind(vname1, vname3, vname6, vname8)`, or the cbind may be used directly, see Example below.

Value

The following are returned as an object to be saved for further use:

- **input**: the name of the input data set
- **xloc**: the vector of locations used for the computations
- **xspread**: the vector of spreads used for the computations
- **ri**: the vector of relative importances provided by the user
- **w**: the vector of weights computed from the relative importances
- **a**: the vector of coefficients - the normalized weights
- **ws**: the computed weighted sums

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see `ltdl.fix.df`. Any rows in the data matrix with with NAs are removed prior to computing the weighted sums.

Author(s)

Robert G. Garrett

References


See Also

- `ltdl.fix.df`, `remove.na`

Examples

```r
## Make test data available
data(sind)
attach(sind)

## Compute weighted sums as in Garrett & Grunsky (2001)
## using medians and interquartile SDs
sind.ws.geea <- wtd.sums(cbind(Zn, Cd, Fe, Mn), ri = c(2, 1, -1, -1),
```
xloc = c(48, 0.6, 1.74, 590),
xspread = c(41.5128, 0.44478, 0.882147, 333.585))

## Compute weighted sums using the median and MAD defaults
sind.ws.def <- wtd.sums(cbind(Zn, Cd, Fe, Mn), ri = c(2, 1, -1, -1))

## Plot the two results against one-another, adding a constant, 
## 3, to the weighted sums to make them positive and log-scale
## plottable
par(pty="s")
plot(sind.ws.gee$ws+3, sind.ws.def$ws+3, log = "xy",
xlim = c(2, 28), ylim = c(2, 28))
abline(0, 1, lty = 3)
abline(v =3, lty = 3)
abline(h = 3, lty = 3)

## Inspect the default weighted sums, adding a constant, 3, to the 
## weighted sums to make them positive and log-scale plottable
shape(sind.ws.def$ws+3, log = TRUE)

## Plot EDA Tukey boxplot based map of default weighted sums
map.eda7(E, N, sind.ws.def$ws)

## Clean-up and detach test data
rm(sind.ws.gee)
rm(sind.ws.def)
par(pty = "m")
detach(sind)

---

**xyplot.eda7**

*Display a Third Variable in a X-Y Plot using Tukey Boxplot Symbology*

**Description**

Displays a third variable where the data are represented by symbols using Tukey boxplot-based symbology. Tukey boxplots divide data into 7 groups, the middle 50%, and three lower and higher groupings, see Details below. The computation of the fences used to subdivide the data may be carried out following a logarithmic transformation of the data. The colours of the symbols may be optionally changed. The x-y plot axes may be optionally displayed with logarithmic (base 10) scaling. Optionally a legend may be added to the plot.

**Usage**

```r
xyplot.eda7(xx, yy, zz, sfact = 1, xlim = NULL, ylim = NULL, log = NULL, logz = FALSE, xlab = deparse(substitute(xx)), ylab = deparse(substitute(yy)), zlab = deparse(substitute(zz)), main = "", ifgrey = FALSE, symcolr = NULL, iflgnd = FALSE, title = deparse(substitute(zz)), ...)
```
Arguments

- **xx**: name of the x-axis variable.
- **yy**: name of the y-axis variable.
- **zz**: name of the third variable to be plotted.
- **sfact**: controls the absolute size of the plotted symbols, by default `sfact = 1`. Increasing `sfact` results in larger symbols.
- **xlim**: user defined limits for the x-axis, see Details below.
- **ylim**: user defined limits for the y-axis, see Details below.
- **log**: if it is required to display the data with logarithmic axis scaling, set `log = "x"` for a logarithmically scaled x-axis, `log = "y"` for a logarithmically scaled y-axis, and `log = "xy"` for both axes logarithmically scaled.
- **logz**: if it is required to undertake the Tukey Boxplot computations after a logarithmic data transform, set `logz = TRUE`.
- **xlab**: by default the character string for `xx` is used for the x-axis title. An alternate title can be displayed with `xlab = "text string"`, see Examples.
- **ylab**: by default the character string for `yy` is used for the y-axis title. An alternate title can be displayed with `ylab = "text string"`, see Examples.
- **zlab**: by default the character string for `zz` is appended to the text string “EDA Tukey Boxplot Based Plot for” for the plot title. An alternate title can be displayed with `zlab = "text string"`, see Details below.
- **main**: an alternative plot title, see Details below.
- **ifgrey**: set `ifgrey = TRUE` if a grey-scale plot is required, see Details below.
- **symcolr**: the default is a colour plot and default colours are provided, deeper blues for lower values, green for the middle 50% of the data, and oranges and reds for higher values. A set of alternate symbol colours can be provided by defining `symcolr`, see Details below.
- **iflgnd**: the default is no legend. If a legend is required set `iflgnd = TRUE`, following the plotting of the data the cursor will be activated, locate that at the top left corner of the desired legend position and ‘left button’ on the pointing device.
- **title**: a short title for the legend, e.g., `title = "Zn (mg/kg)"`. The default is the variable name.
- **...**: further arguments to be passed to methods. For example, if it is required to make the plot title smaller, add `cex.main = 0.9` to reduce the font size by 10%.

Details

Tukey boxplots divide data into 7 groups, the middle 50%, and three lower and higher groupings: within the whisker, near outliers and far outliers, respectively. Symbols for values below the first quartile (Q1) are plotted as increasingly larger circles, while symbols for values above the third quartile are plotted as increasingly larger squares, a ‘+’ is used to plot the data falling in the middle 50%. For the higher groupings, the whisker contains values >Q3 and <=(Q3 + 1.5 * HW), where HW = (Q3 - Q1), the interquartile range; near outliers lie between (Q3 + 1.5 * HW) and (Q3 + 3 * HW); and far outliers have values >Q3 + 3 * HW. For the lower groupings the group
boundaries, fences, fall similarly spaced below Q1. The computation of the fences used to subdivide
the data may be carried out following a logarithmic transformation of the data, set logz = TRUE.
A summary table of the values of the symbol intervals, the number of values plotting as each symbol,
and symbol shapes, sizes and colours is displayed on the current device.

If zlab and main are undefined a default a plot title is generated by appending the input variable
name text string to "EDA Tukey Boxplot-Based Plot for " If no plot title is required set
zlab = "", and if some user defined plot title is required it should be defined in main, e.g.
main = "Plot Title Text".

If the grey-scale option is chosen the symbols are plotted 100% black for the far outliers, 85% black
for the near outliers, 70% black for values within the whiskers, and 60% black for values falling
within the middle 50% of the data.

The default colours, symcolr = c(25, 22, 20, 13, 6, 4, 1), are selected from the
rainbow(36) palette, and alternate colour schemes need to be selected from the same palette.
See display.rainbow for the available colours. It is essential that 7 colours be provided, e.g.,
symcolr = c(27, 24, 22, 12, 5, 3, 36), if exactly 7 are not provided the default colours will
be displayed.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes
representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df.

Any data vectors including NAs are removed prior to displaying the plot.

Author(s)

Robert G. Garrett

See Also

display.rainbow, ltdl.fix.df, remove.na

Examples

```r
## Make test data available
data(sind)attach(sind)

## Plot a default Tukey boxplot-based display
xyplot.eda7(Fe, Mn, Zn)

## Plot with logarithmically scaled boxplot fences and more
## appropriate axis scaling and labelling with a user specified title
xyplot.eda7(Fe, Mn, Zn, sfact = 2, log = "y", logz = TRUE,
xlab = "Fe (ppt) in stream sediment",
ylab = "Mn (mg/kg) in stream sediment",
main = "Howarth & Sinding-Larsen Test Data\nLog10(Zn) (mg/kg)"
)

## Display a grey-scale equivalent of the above plot
xyplot.eda7(Fe, Mn, Zn, sfact = 2, log = "y", logz = TRUE, ifgrey = TRUE,
```
xyplot.eda8

```r
xlab = "Fe (pct) in stream sediment",
ylab = "Mn (mg/kg) in stream sediment",
main = "Howarth & Sinding-Larsen Test Data\nLog\{Zn\} (mg/kg)"

## Plot the same display with an alternate colour scheme
xyplot.eda7(Fe, Mn, Zn, sfact = 2, log = "y", logz = TRUE,
xlab = "Fe (pct) in stream sediment",
ylab = "Mn (mg/kg) in stream sediment",
main = "Howarth & Sinding-Larsen Test Data\nLog\{Zn\} (mg/kg)",
symcolr = c(27, 24, 22, 12, 5, 3, 36))

## Detach test data
detach(sind)
```

### Display a Third Variable in a X-Y Plot as Percentiles

**Description**

Displays a third variable on a X-Y plot where the the third variable is represented by symbols indicating within which group defined by the data’s 2nd, 5th, 25th, 50th, 75th, 95th and 98th percentiles plotted a data value falls. The colours of the symbols may be optionally changed. The x-y plot axes may be optionally displayed with logarithmic (base 10) scaling. Optionally a legend (two options) may be added to the plot.

**Usage**

```r
xyplot.eda8(xx, yy, zz, sfact = 1, xlim = NULL, ylim = NULL,
xlab = deparse(substitute(xx)), ylab = deparse(substitute(yy)),
zlab = deparse(substitute(zz)), main = "", log = NULL,
ifgrey = FALSE, symcolr = NULL, iflgnd = FALSE, pctl = FALSE,
title = deparse(substitute(zz)), ...)
```

**Arguments**

- `xx` name of the x-axis variable.
- `yy` name of the y-axis variable.
- `zz` name of the third variable to be plotted.
- `sfact` controls the absolute size of the plotted symbols, by default `sfact = 1`. Increasing `sfact` results in larger symbols.
- `log` if it is required to display the data with logarithmic axis scaling, set `log = "x"` for a logarithmically scaled x-axis, `log = "y"` for a logarithmically scaled y-axis, and `log = "xy"` for both axes logarithmically scaled.
- `xlim` user defined limits for the x-axis, see Details below.
- `ylim` user defined limits for the y-axis, see Details below.
xlab
by default the character string for xx is used for the x-axis title. An alternate title
 can be displayed with xlab = "text string", see Examples.

ylab
by default the character string for yy is used for the y-axis title. An alternate title
 can be displayed with ylab = "text string", see Examples.

zlab
by default the character string for zz is appended to the text string “EDA Per-
centile Based Plot for” for the plot title. An alternate title can be displayed with
zlab = "text string", see Details below.

main
an alternative plot title, see Details below.

ifgrey
set ifgrey = TRUE if a grey-scale plot is required, see Details below.

symcolr
the default is a colour plot and default colours are provided, deeper blues for
lower values, green for the middle 50% of the data, and oranges and reds for
higher values. A set of alternate symbol colours can be provided by defining
symcolr, see Details below.

iflgnd
the default is no legend. If a legend is required set iflgnd = TRUE, following
the plotting of the data the cursor will be activated, locate that at the top left
corner of the desired legend position and ‘left button’ on the pointing device.
There are two legends to choose from, see pctile below.

pctile
the default legend displays the range of values each symbol represents. Al-
ternately, the percentiles may be displayed rather than their values by setting
pctile = TRUE.

title
a short title for the legend, e.g., title = "Zn (mg/kg)". The default is the
variable name.

... further arguments to be passed to methods. For example, if it is required to make
the plot title smaller, add cex.main = 0.9 to reduce the font size by 10%.

Details
The selected percentiles, 2nd, 5th, 25th, 50th, 75th, 95th and 98th, divide the data into 8 groups.
Values below the median are represented by increasingly larger deeper blue circles below the 25th
percentile (Q1), and values above the 75th percentile (Q3) by increasingly larger orange and red
squares. The mid 50% of the data are represented by green symbols, circles for the median (Q2) to
Q1, and squares for the median (Q2) to Q3.

A summary table of the values of the symbol intervals, the number of values plotting as each symbol,
and symbol shapes, sizes and colours is displayed on the current device.

If zlab and main are undefined a default a plot title is generated by appending the input vari-
able name text string to "EDA Percentile Based Plot for ". If no plot title is required set
zlab = " ", and if some user defined plot title is required it should be defined in main, e.g.
main = "Plot Title Text".

If the grey-scale option is chosen the symbols are plotted 100% black for the far outliers, 85% black
for the near outliers, 70% black for values within the whiskers, and 60% black for values falling
within the middle 50% of the data.

The default colours, symcolr = c(25, 22, 20, 13, 13, 6, 4, 1), are selected from the
rainbow(36) palette, and alternate colour schemes need to be selected from the same palette.
See display.rainbow for the available colours. It is essential that 8 colours be provided, e.g.,
symcolr = c(27, 24, 22, 12, 12, 5, 3, 36), if exactly 8 are not provided the default colours
will be displayed.
Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df.

Any data vectors including NAs are removed prior to displaying the plot.

Author(s)

Robert G. Garrett

See Also

display.rainbow, ltdl.fix.df, remove.na

Examples

```r
## Make test data available
data(sind)
attach(sind)

## Plot a default percentile display
xyplot.eda(Fe, Mn, Zn)

## Plot with more appropriate axis scaling and labelling
## with a user specified title
xyplot.eda(Fe, Mn, Zn, sfact = 2.0, log = "y",
          xlab = "Fe (pct) in stream sediment",
          ylab = "Mn (mg/kg) in stream sediment",
          main = "Howarth & Sinding-Larsen Test Data
Zn (mg/kg)"
)

## Display a grey-scale equivalent of the above plot
xyplot.eda(Fe, Mn, Zn, sfact = 2, log = "y", ifgrey = TRUE,
          xlab = "Fe (pct) in stream sediment",
          ylab = "Mn (mg/kg) in stream sediment",
          main = "Howarth & Sinding-Larsen Test Data
Zn (mg/kg)"
)

## Plot the same display with an alternate colour scheme
xyplot.eda(Fe, Mn, Zn, sfact = 2, log = "y",
          xlab = "Fe (pct) in stream sediment",
          ylab = "Mn (mg/kg) in stream sediment",
          main = "Howarth & Sinding-Larsen Test Data
Zn (mg/kg)",
          symcolr = c(27, 24, 22, 12, 12, 5, 3, 36)
)

## Detach test data
detach(sind)
```
**xyplot.tags**

*Display a Plot of Posted Values for a Third Variable*

**Description**

Displays a x-y plot where the data for a third variable are represented by the ‘written’ values of the data at their x-y position. The x-y plot axes may be optionally displayed with logarithmic (base 10) scaling.

**Usage**

```r
xyplot.tags(xx, yy, tag, log = NULL, xlim = NULL, ylim = NULL,
            xlab = deparse(substitute(xx)), ylab = deparse(substitute(yy)),
            taglab = deparse(substitute(tag)), main = "", ...)```

**Arguments**

- `xx` name of the x-axis variable.
- `yy` name of the y-axis variable.
- `tag` name of the third variable to be displayed.
- `log` if it is required to display the data with logarithmic axis scaling, set `log = "x"` for a logarithmically scaled x-axis, `log = "y"` for a logarithmically scaled y-axis, and `log = "xy"` for both axes logarithmically scaled.
- `xlim` user defined limits for the x-axis, see Details below.
- `ylim` user defined limits for the y-axis, see Details below.
- `xlab` by default the character string for `xx` is used for the x-axis title. An alternate title can be displayed with `xlab = "text string"`, see Examples.
- `ylab` by default the character string for `yy` is used for the y-axis title. An alternate title can be displayed with `ylab = "text string"`, see Examples.
- `taglab` text to be inserted in the plot title, by default `deparse(substitute(tag))` is used. See Details below.
- `main` an alternative plot title, see Details below.
- `...` further arguments to be passed to methods. For example, if smaller plotting characters are required, specify `cex = 0.8`; or if some colour other than black is required for the plotting characters, specify `col = 2` to obtain red (see `display.lty` for the default colour palette). If it is required to make the plot title smaller, add `cex.main = 0.9` to reduce the font size by 10%.

**Details**

If `taglab` and `main` are undefined a default a plot title is generated by appending the input variable name text string to “Plot of Values for”. If no plot title is required set `taglab = " "`, or if an alternative to the variable name `taglab` is required it may be specified, `taglab = "Alternative"`. If some
user defined plot title is required it should be defined in main, e.g., main = "Plot Title Text", in which instance taglab is ignored.

If the default selection for xlim is inappropriate it can be set, e.g., xlim = c(0, 200) or c(2, 200), the latter being appropriate for a logarithmically scaled plot, i.e. log = "x". If the defined limits lie within the observed data range a truncated plot will be displayed. The same procedure applies to setting ylim.

If a plot of sample numbers, ‘IDs’, is required and they are not explicitly in the data frame, a plot of data frame row numbers may be displayed by specifying dimnames(dfname)[[1]] as the value of tags.

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see ltdl.fix.df. Any NAs in the x-y coordinate vectors are removed prior to displaying the plot, thus those ‘data’ are not plotted. However, any NAs in the third variable to be plotted are replaced with a ‘+’ sign to indicate data for the third variable are ‘missing’.

Author(s)

Robert G. Garrett

See Also

ltdl.fix.df, remove.na, display.lty

Examples

```r
## Make test data available
data(sind)
attach(sind)

## Plot the sample site IDs in the x-y space
xyplot(tags(Fe, Mn, ID))

## Plot the data frame row numbers in the x-y space and appropriately
## scale the y-axis
xyplot(tags(Fe, Mn, dimnames(sind)[[1]], log = "y")

## Plot the values for zinc (Zn) in smaller red text in the x-y
## space, providing more appropriate axis scaling and labelling,
## and adding a user specified title
xyplot(tags(Fe, Mn, Zn, log = "y", xlab = "Fe (pct) in stream sediment",
ylab = "Mn (mg/kg) in stream sediment",
main = "Howarth & Sinding-Larsen Test Data\nZn (mg/kg)", cex = 0.8, col = 2)

## Detach test data
detach(sind)
```
Display a Third Variable in a X-Y Plot using Proportional Symbols

Description

Displays a third variable where the data are represented by open circles whose diameters are proportional to the value of the data at their x-y locations. The rate of change of symbol diameter with value and the absolute size of the symbols are defined by the user. The x-y plot axes may be optionally displayed with logarithmic (base 10) scaling. Optionally a legend may be displayed on the plot.

Usage

\[ \text{xyplot.z}(\text{xx}, \text{yy}, \text{zz}, p = 0.5, \text{sfact} = 2.5, \text{zmin} = \text{NA}, \text{zmax} = \text{NA}, \text{log} = \text{NULL}, \text{xlim} = \text{NULL}, \text{ylim} = \text{NULL}, \text{xlab} = \text{deparse(substitute(\text{xx}))}, \text{ylab} = \text{deparse(substitute(\text{yy}))}, \text{zlab} = \text{deparse(substitute(\text{zz}))}, \text{main} = \text{""}, \text{symcolr} = 1, \text{ifparams} = \text{FALSE}, \text{iflgnd} = \text{FALSE}, \text{title} = \text{deparse(substitute(\text{zz}))}, \text{...}) \]

Arguments

- \text{xx} name of the x-axis variable.
- \text{yy} name of the y-axis variable.
- \text{zz} name of the third variable to be plotted.
- \text{p} a parameter that controls the rate of change of symbol diameter with changing value. A default of \text{p} = 0.5 is provided. See Details below.
- \text{sfact} controls the absolute size of the plotted symbols, by default \text{sfact} = 2.5. Increasing \text{sfact} results in larger symbols.
- \text{zmin} a value below which all symbols will be plotted at the same minimum size. By default \text{zmin} = \text{NA} which results in the minimum value of the variable defining the minimum symbol size. See Details below.
- \text{zmax} a value above which all symbols will be plotted at the same maximum size. By default \text{zmax} = \text{NA} which results in the maximum value of the variable defining the maximum symbol size. See Details below.
- \text{log} if it is required to display the data with logarithmic axis scaling, set \text{log} = "x" for a logarithmically scaled x-axis, \text{log} = "y" for a logarithmically scaled y-axis, and \text{log} = "xy" for both axes logarithmically scaled.
- \text{xlim} user defined limits for the x-axis, see Details below.
- \text{ylim} user defined limits for the y-axis, see Details below.
- \text{xlab} by default the character string for \text{xx} is used for the x-axis title. An alternate title can be displayed with \text{xlab} = "text string", see Examples.
- \text{ylab} by default the character string for \text{yy} is used for the y-axis title. An alternate title can be displayed with \text{ylab} = "text string", see Examples.
by default, zlab = deparse(substitute(zz)). A plot title is generated by appending the input variable name text string to "Proportional Symbol Plot for ". Alternative titles may be generated, see Details below.

main

An alternative plot title, see Details below.

symcolr

The colour of the symbols, the default is black, symcolr = 1. This may be changed if required, see display.lty for the default colour palette. For example, symcolr = 2 will cause the symbols to be plotted in red.

ifparams

If ifparams = TRUE on completion of plotting and after the legend has been plotted, if requested, the cursor is activated, locate that at the top left corner of the desired text position and ‘left button’ on the pointing device. This text comprises three lines: the values of p to three significant figures and sfact; the maximum value of z to 3 significant figures and zmax; and the minimum value of z to 3 significant figures and zmin. The default is no text display.

iflgnd

The default is no legend. If a legend is required set iflgnd = TRUE, following the plotting of the data the cursor will be activated, locate that at the top left corner of the desired legend position and ‘left button’ on the pointing device. See Notes below.

title

A short title for the legend, e.g., title = "Zn (mg/kg)". The default is the variable name.

Further arguments to be passed to methods. For example, if smaller plotting characters are required for the legend, specify, for example, cex = 0.8; and if some other colour than black is required for the legend, specify, for example, col = 3, to obtain blue. See display.lty for the default colour palette. If it is required to make the plot title smaller, add cex.main = 0.9 to reduce the font size by 10%.

Details

The symbol diameter is computed as a function of the value z to be plotted:

diameter = dmin + (dmax - dmin) * {(z - zmin)/(zmax - zmin)}^p

where dmin and dmax are defined as 0.1 and 1 units, so the symbol diameters range over an order of magnitude (and symbol areas over two); zmin and zmax are the observed range of the data, or the range over which the user wants the diameters to be computed; and p is a power defined by the user. The value of (z - zmin)/(zmax - zmin) is the value of z normalized, 0 - 1, to the range over which the symbol diameters are to be computed. After being raised to the power p, which will result in a number in the range 0 to 1, this value is multiplied by the permissible range of diameters and added to the minimum diameter. This results in a diameter between 0.1 and 1 units that is proportional to the value of z.

A p value of 1 results in a linear rate of change. Values of p less than unity lead to a rapid initial rate of change with increasing value of z which is often suitable for displaying positively skewed data sets, see the example below. In contrast, values of p greater than unity result in an initial slow rate of change with increasing value of z which is often suitable for displaying negatively skewed data sets. Experimentation is usually necessary to obtain a satisfactory visual effect. See syms.pfunc for a graphic demonstrating the effect of varying the p parameter.

The user may choose to transform the variable to be plotted prior to determining symbol size etc., e.g. log10(zz), to generate a logarithmic rate of symbol size change. See Example below.
If \( z_{\text{min}} \) or \( z_{\text{max}} \) are defined this has the effect of setting a minimum or maximum value of \( z \), respectively, beyond which changes in the value of \( z \) do not result in changes in symbol diameter. This can be useful in limiting the effect of one, or a few, extreme outlier(s) while still plotting them, they simply plot at the minimum or maximum symbol size and are not involved in the calculation of the range of \( z \) over which the symbol diameters vary. Note: If the variable \( z \) includes a transform, e.g., \( \log_{10}(z) \), the values of \( z_{\text{min}} \) and/or \( z_{\text{max}} \) must be in those transform units.

If \( z\text{lab} \) and \( \text{main} \) are undefined a default a plot title is generated by appending the input variable name text string to "Proportional Symbol Plot for " . If no plot title is required set \( z\text{lab} = " " \), and if some user defined plot title is required it should be defined in \( \text{main} \), e.g. \( \text{main} = "\text{Plot Title Text}" \).

If the default selection for \( \text{xlim} \) is inappropriate it can be set, e.g., \( \text{xlim} = \text{c}(0, 200) \) or \( \text{c}(2, 200) \), the latter being appropriate for a logarithmically scaled plot, i.e. \( \log = "x" \). If the defined limits lie within the observed data range a truncated plot will be displayed. The same procedure applies to setting \( \text{ylim} \).

Note

Any less than detection limit values represented by negative values, or zeros or other numeric codes representing blanks in the data, must be removed prior to executing this function, see \texttt{ltdlNfixNdf}.

The legend consists of five proportional symbols and their corresponding \( z \) values: \( z_{\text{min}} \); the three quartiles; and \( z_{\text{max}} \). If \( z_{\text{min}} \) and \( z_{\text{max}} \) have been user defined it is over their range that the symbol sizes are computed and displayed. When defining \( z_{\text{min}} \) and/or \( z_{\text{max}} \) it is useful to set \texttt{ifparams = TRUE} as a reminder, whilst developing the required display.

Any data vectors containing NAs are removed prior to displaying the plot.

Author(s)

Robert G. Garrett

See Also

\texttt{syms, syms.pfunc, ltdlNfixNdf, remove.na, display.lty}

Examples

```r
## Make test data available
data(sind)
attach(sind)

## Display a default symbol plot, \( p = 0.5 \) and \( \text{sfact} = 2.5 \)
xyplot.z(Fe, Mn, Zn)

## Plot with logarithmically scaled symbols and more appropriately
## labelled axes
xyplot.z(Fe, Mn, \log_{10}(Zn), p = 1, \log = "y",
xlab = "Fe (pct) in stream sediment",
ylab = "Mn (mg/kg) in stream sediment")
```
## Plot with differently scaled symbols in red and more appropriate scaling and labelling with a user specified title

```r
xyplot.z(Fe, Mn, Zn, p = 0.3, sfact = 2.0, log = "y",
xlab = "Fe (pct) in stream sediment",
ylab = "Mn (mg/kg) in stream sediment",
main = "Howarth & Sinding-Larsen Test Data\nZn (mg/kg)", symcolr = 2)
```

## Plot as above but where outliers above a value of 1000 displayed with the same symbol

```r
xyplot.z(Fe, Mn, Zn, p = 0.3, sfact = 2.0, zmax = 1000, log = "y",
xlab = "Fe (pct) in stream sediment",
ylab = "Mn (mg/kg) in stream sediment",
main = "Howarth & Sinding-Larsen Test Data\nZn (mg/kg)", symcolr = 2)
```

## Detach test data

```r
detach(sind)
```
Index

*Topic **aplot**
gx.add.chisq, 60
*Topic **arith**
gx.sort, 128
gx.sort.df, 129
*Topic **color**
display.1ty, 44
display.rainbow, 45
gx.cnplots.setup, 64
*Topic **datasets**
ad.test, 12
crm.test, 41
crm.test.new, 42
fix.test, 51
fix.test.asis, 51
kola.c, 153
kola.o, 154
ms.data1, 172
ms.data2, 172
ms.data3, 173
ogrady, 174
ogrady.mat2open, 175
sind, 183
sind.mat2open, 184
triples.test1, 198
triples.test2, 199
*Topic **hplot**
ad.plot1, 5
ad.plot2, 7
ad.plot3, 8
ad.plot4, 10
bwplots, 19
bwplots.by.var, 23
bxplot, 26
caplot, 29
cnplots, 35
crm.plot, 37
crm.plot.new, 39
display.marks, 45
gx.2dproj, 55
gx.2dproj.plot, 58
gx.cnplots, 62
gx.ecdf, 65
gx.hist, 69
gx.ks.test, 72
gx.md.gait, 77
gx.md.gait.closed, 80
gx.md.plot, 83
gx.md.plt0, 85
gx.rqpca.loadplot, 117
gx.rqpca.plot, 118
gx.rqpca.screeplot, 123
inset, 149
inset.exporter, 151
map.eda7, 161
map.eda8, 164
map.tags, 166
map.z, 168
shape, 180
syms, 186
syms.pfunc, 187
tbplots, 188
tbplots.by.var, 191
thplot1, 194
thplot2, 196
xyplot.eda7, 204
xyplot.eda8, 207
xyplot.tags, 210
xyplot.z, 212
*Topic **htest**
ad.plot3, 8
ad.plot4, 10
anova1, 16
anova2, 18
gx.hypergeom, 71
gx.ks.test, 72
gx.pearson, 102
gx.rma, 106
INDEX

gx.runs, 124
gx.spearman, 130
gx.triples.aov, 142
gx.triples.fgx, 143
thplot1, 194
thplot2, 196
*Topic iplot
  gx.md.gait, 77
gx.md.gait.closed, 80
*Topic manip
  alr, 13
  alts2dups, 15
clr, 33
expit, 46
ilr, 146
logit, 155
orthonorm, 176
rng, 179
*Topic misc
  alts2dups, 15
cat2list, 32
cutter, 42
df.test, 43
display.ascii.o, 44
display.lty, 44
display.marks, 45
display.rainbow, 45
gx.cnplts.setup, 64
gx.sort, 128
gx.sort.df, 129
gx.subset, 133
ltdl.fix, 157
ltdl.fix.df, 159
remove.na, 178
var2fact, 200
where.na, 201
*Topic models
  gx.adj.r2, 61
gx.lm.vif, 74
*Topic multivariate
  alr, 13
clr, 33
gx.2dproj, 55
gx.md.gait, 77
gx.md.gait.closed, 80
gx.mva, 88
gx.mva.closed, 91
gx.mvalloc, 94
gx.mvalloc.closed, 96
gx.pairs4parts, 100
gx.pearson, 102
gx.plot2parts, 103
gx.robmva, 108
gx.robmva.closed, 112
gx.rotate, 115
gx.scores, 125
gx.sm, 126
gx.spearman, 130
gx.vm, 145
ilr, 146
ilr.stab, 148
orthonorm, 176
wtd.sums, 202
*Topic nonparametric
  gx.ks.test, 72
gx.runs, 124
gx.spearman, 130
*Topic package
  rgr_1.1.10-package, 4
*Topic print
  gx.md.display, 76
gx.md.print, 86
gx.mvalloc.print, 99
gx.rqplca.print, 121
*Topic regression
  gx.rma, 106
*Topic robust
  gx.robmva, 108
gx.robmva.closed, 112
gx.spearman, 130
*Topic univariate
  expit, 46
logit, 155
*Topic univar
  ad.plot1, 5
ad.plot2, 7
ad.plot3, 8
ad.plot4, 10
anova1, 16
anova2, 18
crm.plot, 37
crm.plot.new, 39
fences, 47
fences.summary, 49
framework.stats, 52
framework.summary, 53
INDEX

alts2dups, 15
alr, 13, 33, 46, 47, 147, 156
alts2dups, 15
anova, 6, 10, 16, 18, 19, 157, 159, 172
anova2, 16–18, 18, 172, 173

bplots, 19, 26, 27, 32, 48, 50, 137
bplots.by.var, 23, 138, 200
bxplot, 26, 47, 49, 101, 104, 182
caplot, 29, 35, 48, 49
cat2list, 22, 32, 190
cbind, 25, 193
clr, 13, 14, 33, 34, 46, 47, 89, 103, 110, 131, 147, 156, 177
cmdscale, 57
cnpplt, 31, 35, 47, 49, 151, 182
colors, 30, 31
crm.plot, 37, 40
crm.plot.new, 39, 42
crm.test, 41
crm.test.new, 42
cutter, 42
df.test, 43
display.ascii.o, 44
display.ity, 9, 11, 21, 24, 27, 28, 44, 56, 58, 63, 65, 70, 73, 78, 81, 84, 86, 119, 123, 149, 167–170, 181, 182, 189, 192, 210, 211, 213, 214
display.marks, 21, 25, 35, 36, 45, 63, 65–67, 73, 150, 152, 181, 182
display.rainbow, 45, 163, 165, 166, 206, 208, 209
display.rainbow, 124, 125
expit, 46, 48, 156
fastICA, 57
fences, 47, 50, 71
fences.summary, 47, 48, 49
fix.test, 51, 52
fix.test.asis, 51, 51
framework.stats, 52, 54
framework.summary, 22, 52, 53, 137, 190
gx.2dproj, 55, 58, 59
gx.2dproj.plot, 57, 58
gx.add.chisq, 60, 84, 86
gx.adjr2, 61
gx.cnpplets, 48, 50, 62, 73
gx.cnpplets.setup, 63, 64
gx.ecdf, 47, 49, 65, 104, 182
gx.fractile, 67, 105
gx.hist, 27, 35, 47, 49, 66, 69, 151, 152, 181, 182
gx.hypergeom, 71, 124
gx.ks.test, 72
gx.lm.vif, 74
gx.md.display, 76
gx.md.gait, 61, 76, 77, 77, 83–87, 94, 95, 110
gx.md.gait.closed, 61, 76, 77, 80, 83, 84, 86, 94, 95, 97, 98, 177
gx.md.plot, 61, 78, 79, 81, 83, 85, 86, 88, 90, 91, 93, 108, 109, 111–114
gx.md.plot0, 83, 85
gx.mva.closed, 77, 87, 94, 99
gx.mvalloc, 81, 96, 99
gx.mvalloc.print, 76, 77, 87, 95, 97, 98, 99
gx.pairs4parts, 100, 102, 131
gx.pearson, 102
gx.plot2parts, 102, 103, 131
gx.quantile, 68, 105
gx.rma, 106
sind.mat2open, 103, 184
split, 21, 32, 189
step, 61
subset, 134
summary, 61
syms, 171, 186, 214
syms.pfunc, 170, 171, 186, 187, 187, 213, 214
tbplots, 27, 32, 48, 50, 137, 188, 193
tbplots.by.var, 138, 191, 200
text, 63, 73
thplot1, 172, 194, 196–198
thplot2, 172, 173, 195, 196, 196
triples.test1, 142–144, 198
triples.test2, 143, 144, 199
var2fact, 26, 193, 200
varimax, 116
where.na, 89, 93, 110, 114, 178, 201
wtd.sums, 202
xyplot.eda7, 204
xyplot.eda8, 207
xyplot.tags, 120, 210
xyplot.z, 212