

# Package ‘StatDA’

July 21, 2025

**Version** 1.7.11

**Type** Package

**Title** Statistical Analysis for Environmental Data

**Maintainer** Peter Filzmoser <Peter.Filzmoser@tuwien.ac.at>

**Description** Statistical analysis methods for environmental data are implemented. There is a particular focus on robust methods, and on methods for compositional data. In addition, larger data sets from geochemistry are provided. The statistical methods are described in Reimann, Filzmoser, Garrett, Dutter (2008, ISBN:978-0-470-98581-6).

**Depends** R (>= 2.10), methods, sgeostat

**Imports** cluster, e1071, MASS, MBA, mgcv, rgl, robustbase, xtable, sp, geoR

**Suggests** mclust

**License** GPL (>= 3)

**URL** <http://cstat.tuwien.ac.at/filz/>

**Author** Peter Filzmoser [aut, cre, cph]

**Repository** CRAN

**Date/Publication** 2023-06-02 12:20:05 UTC

**NeedsCompilation** no

## Contents

arw . . . . .	3
AuNEW . . . . .	4
AuOLD . . . . .	5
bhorizon . . . . .	5
bordersKola . . . . .	8
boxes . . . . .	9
boxplotlegend . . . . .	11
boxplotlog . . . . .	12
boxplotperc . . . . .	14
bubbleFIN . . . . .	16

CHorANADUP . . . . .	17
CHorFieldDUP . . . . .	23
chorizon . . . . .	30
CHorSTANDARD . . . . .	34
concarea . . . . .	37
concareaExampleKola . . . . .	39
cor.sign . . . . .	41
CorCompare . . . . .	42
CorGroups . . . . .	43
do.ellipses . . . . .	44
edaplot . . . . .	45
edaplotlog . . . . .	47
factanal.fit.principal . . . . .	48
kola.background . . . . .	49
KrigeLegend . . . . .	50
loadplot . . . . .	52
monch . . . . .	53
moss . . . . .	54
nizap . . . . .	57
Northarrow . . . . .	57
ohorizon . . . . .	58
pfa . . . . .	62
plotbg . . . . .	63
plotelement . . . . .	64
plotellipse . . . . .	65
plotmvoutlier . . . . .	66
plotuniout . . . . .	67
polygrid . . . . .	69
polys . . . . .	70
ppplot.das . . . . .	72
qpplot.das . . . . .	73
qqplot.das . . . . .	74
res.eyefit.As_C . . . . .	76
res.eyefit.As_C_m . . . . .	76
res.eyefit.AuNEW . . . . .	77
res.eyefit.Ca_C . . . . .	78
res.eyefit.Ca_O . . . . .	78
res.eyefit.Hg_O . . . . .	79
res.eyefit.Pb_O1 . . . . .	80
res.eyefit.Pb_O2 . . . . .	80
rg.boxplot . . . . .	81
rg.mva . . . . .	82
rg.mvalloc . . . . .	84
rg.remove.na . . . . .	85
rg.robmva . . . . .	86
rg.wtdsums . . . . .	88
RobCor.plot . . . . .	89
roundpretty . . . . .	90

roundpretty.sub . . . . .	91
scalebar . . . . .	92
scatter3dPETER . . . . .	93
SmoothLegend . . . . .	95
suns . . . . .	97
SymbLegend . . . . .	98
ternary . . . . .	100
timetrend . . . . .	101
topsoil . . . . .	103
varcomp . . . . .	105

## Index 107

---

arw	<i>Adaptive reweighted estimator for multivariate location and scatter</i>
-----	--

---

### Description

Adaptive reweighted estimator for multivariate location and scatter with hard-rejection weights. The multivariate outliers are defined according to the supremum of the difference between the empirical distribution function of the robust Mahalanobis distance and the theoretical distribution function.

### Usage

```
arw(x, m0, c0, alpha, pcrit)
```

### Arguments

x	Dataset (n x p)
m0	Initial location estimator (1 x p)
c0	Initial scatter estimator (p x p)
alpha	Maximum thresholding proportion (optional scalar, default: alpha = 0.025)
pcrit	Critical value obtained by simulations (optional scalar, default value obtained from simulations)

### Details

At the basis of initial estimators of location and scatter, the function arw performs a reweighting step to adjust the threshold for outlier rejection. The critical value pcrit was obtained by simulations using the MCD estimator as initial robust covariance estimator. If a different estimator is used, pcrit should be changed and computed by simulations for the specific dimensions of the data x.

### Value

m	Adaptive location estimator (p x 1)
c	Adaptive scatter estimator (p x p)
cn	Adaptive threshold ("adjusted quantile")
w	Weight vector (n x 1)

**Author(s)**

Moritz Gschwandtner <<e0125439@student.tuwien.ac.at>>  
Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

P. Filzmoser, R.G. Garrett, and C. Reimann (2005). Multivariate outlier detection in exploration geochemistry. *Computers & Geosciences*, 31:579-587.

**Examples**

```
x <- cbind(rnorm(100), rnorm(100))  
arw(x, apply(x,2,mean), cov(x))
```

---

AuNEW

*Au data, new*

---

**Description**

Au data from Kola C-horizon, new measurement method

**Usage**

```
data(AuNEW)
```

**Format**

The format is: num [1:606] 0.001344 0.000444 0.001607 0.000713 0.000898 ...

**Details**

These data of Au have much higher quality than the data AuOLD.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
data(AuNEW)  
data(AuOLD)  
plot(log10(AuOLD), log10(AuNEW))
```

---

AuOLD

*Au data, old*

---

**Description**

Au data from Kola C-horizon, old measurement method

**Usage**

```
data(AuOLD)
```

**Format**

The format is: num [1:606] 0.001 0.001 0.002 0.001 0.007 0.006 0.001 0.001 0.001 0.001 ...

**Details**

These data of Au have much worse quality than the data AuNEW.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
data(AuNEW)
data(AuOLD)
plot(log10(AuOLD), log10(AuNEW))
```

---

bhorizon

*B-horizon of the Kola Data*

---

**Description**

The Kola Data were collected in the Kola Project (1993-1998, Geological Surveys of Finland (GTK) and Norway (NGU) and Central Kola Expedition (CKE), Russia). More than 600 samples in five different layers were analysed, this dataset contains the B-horizon.

**Usage**

```
data(bhorizon)
```

**Format**

A data frame with 609 observations on the following 77 variables.

ID a numeric vector

XCOO a numeric vector

YCOO a numeric vector

ELEV a numeric vector

COUN a factor with levels FIN NOR RUS

ASP a factor with levels E FLAT N NE NW NW S SE SW W

LOWDB a numeric vector

LITO a numeric vector

GENLAN a factor with levels DEEPVAL FLA PLAIN FLAT HIMO LOWMO PLAIN PLAT RIDGE SLOPE

Ag a numeric vector

Al a numeric vector

Al\_XRF a numeric vector

Al203 a numeric vector

As a numeric vector

Au a numeric vector

B a numeric vector

Ba a numeric vector

Be a numeric vector

Bi a numeric vector

Br\_IC a numeric vector

Ca a numeric vector

Ca\_XRF a numeric vector

CaO a numeric vector

Cd a numeric vector

Cl\_IC a numeric vector

Co a numeric vector

Cr a numeric vector

Cu a numeric vector

EC a numeric vector

F\_IC a numeric vector

Fe a numeric vector

Fe\_XRF a numeric vector

Fe203 a numeric vector

Hg a numeric vector

K a numeric vector

K\_XRF a numeric vector  
K2O a numeric vector  
La a numeric vector  
Li a numeric vector  
LOI a numeric vector  
Mg a numeric vector  
Mg\_XRF a numeric vector  
MgO a numeric vector  
Mn a numeric vector  
Mn\_XRF a numeric vector  
MnO a numeric vector  
Mo a numeric vector  
Na a numeric vector  
Na\_XRF a numeric vector  
Na2O a numeric vector  
Ni a numeric vector  
NO3\_IC a numeric vector  
P a numeric vector  
P\_XRF a numeric vector  
P2O5 a numeric vector  
Pb a numeric vector  
Pd a numeric vector  
pH a numeric vector  
PO4\_IC a numeric vector  
Pt a numeric vector  
S a numeric vector  
Sb a numeric vector  
Sc a numeric vector  
Se a numeric vector  
Si a numeric vector  
Si\_XRF a numeric vector  
SiO2 a numeric vector  
SO4\_IC a numeric vector  
Sr a numeric vector  
Te a numeric vector  
Th a numeric vector  
Ti a numeric vector  
Ti\_XRF a numeric vector  
TiO2 a numeric vector  
V a numeric vector  
Y a numeric vector  
Zn a numeric vector

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**Source**

Kola Project (1993-1998)

**References**

Reimann C, Jørgensen M, Chekushin V, Bogatyrev I, Boyd R, Caritat P de, Dutter R, Finne TE, Halleraker JH, Jørgensen M, Kashulina G, Lehto O, Niskavaara H, Pavlov V, Rönkä ML, Strand T, Volden T. Environmental Geochemical Atlas of the Central Barents Region. NGU-GTK-CKE Special Publication, Geological Survey of Norway, Trondheim, Norway, 1998.

**Examples**

```
data(bhorizon)
str(bhorizon)
```

---

bordersKola

*Borders of the Kola Project boundary*

---

**Description**

x- and y-coordinates of the Kola Project boundary.

**Usage**

```
data(bordersKola)
```

**Format**

The format is: List of 2 \$ x: num [1:64] 836200 881000 752900 743100 737500 ... \$ y: num [1:64] 7708867 7403003 7389239 7377769 7364006 ...

**Details**

The coordinates for the Kola Project boundary are used for the surface maps, i.e. for Krige and Smoothing maps. It is a list with two list elements x and y for the x- and y-coordinates.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.



**Examples**

```
data(bordersKola)
plot(bordersKola$x,bordersKola$y)
```

boxes

*Boxes***Description**

The function `boxes` computes boxes of multivariate data. If `add=TRUE` the boxes are plotted in the current plot otherwise nothing is plotted.

**Usage**

```
boxes(x, xA = 1, yA = 2, zA = 3, labels = dimnames(x)[[1]], locations = NULL,
      nrow = NULL, ncol = NULL, key.loc = NULL, key.labels = dimnames(x)[[2]],
      key.xpd = TRUE, xlim = NULL, ylim = NULL, flip.labels = NULL, len = 1,
      leglen = 1, axes = FALSE, frame.plot = axes, main = NULL, sub = NULL,
      xlab = "", ylab = "", cex = 0.8, lwd = 0.25, lty = par("lty"), xpd = FALSE,
      mar = pmin(par("mar"), 1.1 + c(2 * axes + (xlab != ""), 2 * axes + (ylab != "")),
      1, 0)), add = FALSE, plot = TRUE, ...)
```

**Arguments**

<code>x</code>	multivariate data in form of matrix or data frame
<code>xA</code>	assignment of clusters to the coordinates of the boxes
<code>yA</code>	assignment of clusters to the coordinates of the boxes
<code>zA</code>	assignment of clusters to the coordinates of the boxes
<code>labels</code>	vector of character strings for labeling the plots
<code>locations</code>	locations for the boxes on the plot (e.g. X/Y coordinates)
<code>nrow</code>	integers giving the number of rows and columns to use when 'locations' is 'NULL'. By default, 'nrow == ncol', a square will be used.
<code>ncol</code>	integers giving the number of rows and columns to use when 'locations' is 'NULL'. By default, 'nrow == ncol', a square will be used.
<code>key.loc</code>	vector with x and y coordinates of the unit key.
<code>key.labels</code>	vector of character strings for labeling the segments of the unit key. If omitted, the second component of 'dimnames(x)' is used, if available.
<code>key.xpd</code>	clipping switch for the unit key (drawing and labeling), see 'par("xpd")'.
<code>xlim</code>	vector with the range of x coordinates to plot
<code>ylim</code>	vector with the range of y coordinates to plot
<code>flip.labels</code>	logical indicating if the label locations should flip up and down from diagram to diagram. Defaults to a somewhat smart heuristic.
<code>len</code>	multiplicative values for the space used in the plot window

<code>leglen</code>	multiplicative values for the space of the labels on the legend
<code>axes</code>	logical flag: if 'TRUE' axes are added to the plot
<code>frame.plot</code>	logical flag: if 'TRUE', the plot region ist framed
<code>main</code>	a main title for the plot
<code>sub</code>	a sub title for the plot
<code>xlab</code>	a label for the x axis
<code>ylab</code>	a label for the y axis
<code>cex</code>	character expansion factor for the labels
<code>lwd</code>	line width used for drawing
<code>lty</code>	line type used for drawing
<code>xpd</code>	logical or NA indicationg if clipping should be done, see 'par(xpd=.)'
<code>mar</code>	argument to 'par(mar=*)', rypically choosing smaller margings than by default
<code>add</code>	logical, if 'TRUE' add boxes to current plot
<code>plot</code>	logical, if 'FALSE', nothing is plotted
<code>...</code>	further arguments, passed to the first call of 'plot()'

### Details

This type of graphical approach for multivariate data is only applicable where the data can be grouped into three clusters. This means that before the plot can be made the data undergo a hierarchical cluster to get the size of each cluster. The distance measure for the hierarchicla cluster is complete linkage. Each cluster represents one side of the boxes.

### Value

No return value, creates a plot.

### Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

### References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

### See Also

[plot.default,box](#)

**Examples**

```
#plots the background and the boxes for the elements
data(ohorizon)
X=ohorizon[,"XC00"]
Y=ohorizon[,"YC00"]
el=log10(ohorizon[,c("Co","Cu","Ni","Rb","Bi","Na","Sr")])
data(kola.background)

sel <- c(3,8,22, 29, 32, 35, 43, 69, 73 ,93,109,129,130,134,168,181,183,205,211,
        218,237,242,276,292,297,298,345,346,352,372,373,386,408,419,427,441,446,490,
        516,535,551,556,558,564,577,584,601,612,617)

x=el[sel,]
xwid=diff(range(X))/12e4
ywid=diff(range(Y))/12e4
plot(X,Y,frame.plot=FALSE,xaxt="n",yaxt="n",xlab="",ylab="",type="n",
     xlim=c(360000,max(X)))
plotbg(map.col=c("gray","gray","gray","gray"),add.plot=TRUE)

boxes(x,locations=cbind(X[sel],Y[sel]),len=20000,key.loc=c(800000,7830000),leglen=25000,
      cex=0.75, add=TRUE, labels=NULL, lwd=1.1)
```

boxplotlegend

*Boxplotlegend***Description**

This function plots the legend in form of a boxplot. The symbols represent the different levels (e.g. whiskers, median, ...) of the boxplot.

**Usage**

```
boxplotlegend(X, Y, el, boxinfo, x.shift = 40000, xf = 10000, y.shift = 0.2,
             y.scale = 130000, legend.title = "Legend", cex.legtit = 1, logscale = TRUE,
             symb = c(1, 1, 16, 3, 3), ssize = c(1.5, 1, 0.3, 1, 1.5), accentuate = FALSE,
             cex.scale = 0.8)
```

**Arguments**

X	X-coordinates
Y	Y-coordinates
el	variable considered
boxinfo	from boxplot(el) or boxplotlog(el)
x.shift	shift in x-direction
xf	width in x-direction
y.shift	shift in y-direction (from title)

<code>y.scale</code>	scale in y-direction
<code>legend.title</code>	title for legend
<code>cex.legtit</code>	cex of title for legend
<code>logscale</code>	if TRUE plot boxplot in log-scale
<code>symb</code>	symbols to be used (length 5!)
<code>ssize</code>	symbol sizes to be used (length 5!)
<code>accentuate</code>	if FALSE no symbols for the upper values (e.g. upper "hinge", upper whisker) are assigned
<code>cex.scale</code>	cex for text "log-scale" for scale

### Details

Takes the information provided by the argument `boxinfo` and plots a boxplot corresponding to the values. If there are no upper or/and lower outliers the symbols for the upper or/and lower whiskers will be ignored.

### Value

Plots the legend with respect to the boxplot and returns the symbols, size and the quantiles used for the legend.

### Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

### References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

### Examples

```
#internal function, used in SymbLegend
```

---

`boxplotlog`

*Boxplotlog*

---

### Description

The function `boxplot` plots a boxplot of the data with respect to the logarithmic transformed values of the whiskers. See also details.

### Usage

```
boxplotlog(x, ..., range = 1.5, width = NULL, varwidth = FALSE, notch = FALSE,
outline = TRUE, names, plot = TRUE, border = par("fg"), col = NULL, log = "",
pars = list(boxwex = 0.8, staplewex = 0.5, outwex = 0.5), horizontal = FALSE,
add = FALSE, at = NULL)
```

**Arguments**

x	data
...	further arguments for creating the list
range	this determines how far the plot "whiskers" extend from the box. If range is positive, the most extreme data point which is no more than range times the length of the box away from the box. A value of zero causes the whiskers to extend to the data extremes.
width	a vector giving the relative widths of the boxes making up the plot
varwidth	if varwidth is TRUE, the boxes are drawn with widths proportional to the square-roots of the number of observations in the groups.
notch	if notch is TRUE, a notch is drawn in each side of the boxes
outline	if outline is FALSE, the outliers are not drawn
names	define the names of the attributes
plot	if plot is TRUE the boxplot is plotted in the current plot
border	character or numeric (vector) which indicates the color of the box borders
col	defines the colour
log	character, indicating if any axis should be drawn in logarithmic scale
pars	some graphical parameters can be specified
horizontal	logical parameter indicating if the boxplots should be horizontal; FALSE means vertical boxes
add	if TRUE the boxplot is added to the current plot
at	numeric vector giving the locations of the boxplots

**Details**

Sometimes a boxplot of the original data does not identify outliers because the boxplot assumes normal distribution. Therefore the data are logarithmically transformed and values for plotting the boxplot are calculated. After that the data are backtransformed and the boxplot is plotted with respect to the logarithmic results. Now the outliers are identified.

**Value**

stats	a vector of length 5, containing the extreme of the lower whisker, the lower "hinge", the median, the upper "hinge" and the extreme of the upper whisker (backtransformed)
n	the number of non-NA observations in the sample
conf	the lower and upper extremes of the "notch"
out	the values of any data points which lie beyond the extremes of the whiskers (backtransformed)
group	the group
names	the attributes

Returns a boxplot which is calculated with the log-transformed data.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
data(chorizon)
Ba=chorizon[,"Ba"]

boxplotlog((Ba),horizontal=TRUE,xlab="Ba [mg/kg]",cex.lab=1.4,pch=3,cex=1.5)
```

---

boxplotperc

*Boxplot based on percentiles*

---

**Description**

This function plots a boxplot of the data and the boundaries are based on percentiles.

**Usage**

```
boxplotperc(x, ..., quant = c(0.02, 0.98), width = NULL, varwidth = FALSE,
notch = FALSE, outline = TRUE, names, plot = TRUE, border = par("fg"),
col = NULL, log = "", pars = list(boxwex = 0.8, staplewex = 0.5, outwex = 0.5),
horizontal = FALSE, add = FALSE, at = NULL)
```

**Arguments**

x	data
...	further arguments for creating the list
quant	the underlying percentages
width	a vector giving the relative widths of the boxes making up the plot
varwidth	if varwidth is TRUE, the boxes are drawn with widths proportional to the square-roots of the number of observations in the groups.
notch	if notch is TRUE, a notch is drawn in each side of the boxes
outline	if outliers is FALSE, the outliers are not drawn
names	define the names of the attributes
plot	if plot is TRUE the boxplot is plotted in the current plot
border	character or numeric (vector) which indicates the color of the box borders
col	defines the colour
log	character, indicating if any axis should be drawn in logarithmic scale

<code>pars</code>	some graphical parameters can be specified
<code>horizontal</code>	logical parameter indicating if the boxplots should be horizontal; FALSE means vertical boxes
<code>add</code>	if TRUE the boxplot is added to the current plot
<code>at</code>	numeric vector giving the locations of the boxplots

### Details

The default value for `quant` is the 2% and 98% quantile and this argument defines the percentiles for the upper and lower whiskers.

### Value

<code>stats</code>	a vector of length 5, containing the extreme of the lower whisker, the lower "hinge", the median, the upper "hinge" and the extreme of the upper whisker (backtransformed)
<code>n</code>	the number of non-NA observations in the sample
<code>conf</code>	the lower and upper extremes of the "notch"
<code>out</code>	the values of any data points which lie beyond the extremes of the whiskers (backtransformed)
<code>group</code>	the group
<code>names</code>	the attributes

### Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

### References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

### See Also

[boxplotlog](#)

### Examples

```
data(chorizon)
Ba=chorizon[, "Ba"]
boxplotperc(Ba, quant=c(0.05, 0.95), horizontal=TRUE, xlab="Ba [mg/kg]", cex.lab=1.2, pch=3)
```

---

 bubbleFIN

*Bubbleplot due to Finnish method*


---

**Description**

This function plots multivariate data with respect to the value. The size of the bubble represents the value of the datapoint.

**Usage**

```
bubbleFIN(x, y, z, radi = 10000, S = 9, s = 0.9, wa = 0, wb = 0.95, wc = 0.05,
  plottitle = "BubblePlot", legendtitle = "Legend", text.cex = 1,
  legtitle.cex = 1, backgr = "kola.background", leg = TRUE, ndigits = 1)
```

**Arguments**

x	x coordinates
y	y coordinates
z	measured value at point (x,y)
radi	scaling for the map
S, s	control the size of the largest and smallest bubbles
wa, wb, wc	factors which defines the shape of the exponential function
plottitle	the titel of the plot
legendtitle	the titel of the legend
text.cex	multiplier for the size of the labels
legtitle.cex	multiplier for the size of the legendtitle
backgr	which background should be used
leg	if TRUE the bubbles are plotted to the legend
ndigits	how much digits should be plotted at the legend

**Details**

The smallest bubbles represent the 10% quantile and the biggest bubbles represent the 99

**Value**

Plots bubbles in the existing plot.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>



## References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

## Examples

```
data(kola.background)
data(ohorizon)
el=ohorizon[, "Mg"]
X=ohorizon[, "XC00"]
Y=ohorizon[, "YC00"]
plot(X,Y,frame.plot=FALSE,xaxt="n",yaxt="n",xlab="",ylab="",type="n") #plot bubbles with background
plotbg(map.col=c("gray", "gray", "gray", "gray"),add.plot=TRUE)

bubbleFIN(X,Y,el,S=9,s=2,plottitle="",legendtitle="Mg [mg/kg]", text.cex=0.63,legtitle.cex=0.80)
```

---

CHorANADUP

*Analytical duplicates of the C-horizon Kola data*

---

## Description

Analytical duplicates have been selected for quality control.

## Usage

```
data(CHorANADUP)
```

## Format

A data frame with 52 observations on the following 190 variables.

A1\_.Loc a numeric vector  
 A2\_.Loc a numeric vector  
 A1\_Ag a numeric vector  
 A1\_Ag\_INAA a numeric vector  
 A1\_A1 a numeric vector  
 A1\_A1203 a numeric vector  
 A1\_As a numeric vector  
 A1\_As\_INAA a numeric vector  
 A1\_Au\_INAA a numeric vector  
 A1\_B a numeric vector  
 A1\_Ba a numeric vector  
 A1\_Ba\_INAA a numeric vector  
 A1\_Be a numeric vector

A1\_Bi a numeric vector  
A1\_Br a numeric vector  
A1\_Br\_INAA a numeric vector  
A1\_Ca a numeric vector  
A1\_Ca\_INAA a numeric vector  
A1\_CaO a numeric vector  
A1\_Cd a numeric vector  
A1\_Ce\_INAA a numeric vector  
A1\_Cl a numeric vector  
A1\_Co a numeric vector  
A1\_Co\_INAA a numeric vector  
A1\_Cond a numeric vector  
A1\_Cr a numeric vector  
A1\_Cr\_INAA a numeric vector  
A1\_Cs\_INAA a numeric vector  
A1\_Cu a numeric vector  
A1\_Eu\_INAA a numeric vector  
A1\_F a numeric vector  
A1\_F\_ionselect a numeric vector  
A1\_Fe a numeric vector  
A1\_Fe\_INAA a numeric vector  
A1\_Fe203 a numeric vector  
A1\_Hf\_INAA a numeric vector  
A1\_Hg a numeric vector  
A1\_Hg\_INAA a numeric vector  
A1\_Ir\_INAA a numeric vector  
A1\_K a numeric vector  
A1\_K20 a numeric vector  
A1\_La a numeric vector  
A1\_La\_INAA a numeric vector  
A1\_Li a numeric vector  
A1\_LOI a numeric vector  
A1\_Lu\_INAA a numeric vector  
A1\_Mass\_INAA a numeric vector  
A1\_Mg a numeric vector  
A1\_MgO a numeric vector  
A1\_Mn a numeric vector

A1\_Mn0 a numeric vector  
A1\_Mo a numeric vector  
A1\_Mo\_INAA a numeric vector  
A1\_Na a numeric vector  
A1\_Na\_INAA a numeric vector  
A1\_Na20 a numeric vector  
A1\_Nd\_INAA a numeric vector  
A1\_Ni a numeric vector  
A1\_Ni\_INAA a numeric vector  
A1\_N02 a numeric vector  
A1\_N03 a numeric vector  
A1\_P a numeric vector  
A1\_P205 a numeric vector  
A1\_Pb a numeric vector  
A1\_pH a numeric vector  
A1\_P04 a numeric vector  
A1\_Rb a numeric vector  
A1\_S a numeric vector  
A1\_Sb a numeric vector  
A1\_Sb\_INAA a numeric vector  
A1\_Sc a numeric vector  
A1\_Sc\_INAA a numeric vector  
A1\_Se a numeric vector  
A1\_Se\_INAA a numeric vector  
A1\_Si a numeric vector  
A1\_SiO2 a numeric vector  
A1\_Sm\_INAA a numeric vector  
A1\_Sn\_INAA a numeric vector  
A1\_S04 a numeric vector  
A1\_Sr a numeric vector  
A1\_Sr\_INAA a numeric vector  
A1\_Sum a numeric vector  
A1-Ta\_INAA a numeric vector  
A1\_Tb\_INAA a numeric vector  
A1\_Te a numeric vector  
A1\_Th a numeric vector  
A1\_Th\_INAA a numeric vector

A1\_Ti a numeric vector  
A1\_TiO2 a numeric vector  
A1\_U\_INAA a numeric vector  
A1\_V a numeric vector  
A1\_W\_INAA a numeric vector  
A1\_Y a numeric vector  
A1\_Yb\_INAA a numeric vector  
A1\_Zn a numeric vector  
A1\_Zn\_INAA a numeric vector  
A2\_Ag a numeric vector  
A2\_Ag\_INAA a numeric vector  
A2\_Al a numeric vector  
A2\_Al2O3 a numeric vector  
A2\_As a numeric vector  
A2\_As\_INAA a numeric vector  
A2\_Au\_INAA a numeric vector  
A2\_B a numeric vector  
A2\_Ba a numeric vector  
A2\_Ba\_INAA a numeric vector  
A2\_Be a numeric vector  
A2\_Bi a numeric vector  
A2\_Br a numeric vector  
A2\_Br\_INAA a numeric vector  
A2\_Ca a numeric vector  
A2\_Ca\_INAA a numeric vector  
A2\_CaO a numeric vector  
A2\_Cd a numeric vector  
A2\_Ce\_INAA a numeric vector  
A2\_Cl a numeric vector  
A2\_Co a numeric vector  
A2\_Co\_INAA a numeric vector  
A2\_Cond a numeric vector  
A2\_Cr a numeric vector  
A2\_Cr\_INAA a numeric vector  
A2\_Cs\_INAA a numeric vector  
A2\_Cu a numeric vector  
A2\_Eu\_INAA a numeric vector

A2\_F a numeric vector  
A2\_F\_ionselect a numeric vector  
A2\_Fe a numeric vector  
A2\_Fe\_INAA a numeric vector  
A2\_Fe203 a numeric vector  
A2\_Hf\_INAA a numeric vector  
A2\_Hg a numeric vector  
A2\_Hg\_INAA a numeric vector  
A2\_Ir\_INAA a numeric vector  
A2\_K a numeric vector  
A2\_K20 a numeric vector  
A2\_La a numeric vector  
A2\_La\_INAA a numeric vector  
A2\_Li a numeric vector  
A2\_LOI a numeric vector  
A2\_Lu\_INAA a numeric vector  
A2\_Mass\_INAA a numeric vector  
A2\_Mg a numeric vector  
A2\_MgO a numeric vector  
A2\_Mn a numeric vector  
A2\_MnO a numeric vector  
A2\_Mo a numeric vector  
A2\_Mo\_INAA a numeric vector  
A2\_Na a numeric vector  
A2\_Na\_INAA a numeric vector  
A2\_Na20 a numeric vector  
A2\_Nd\_INAA a numeric vector  
A2\_Ni a numeric vector  
A2\_Ni\_INAA a numeric vector  
A2\_N02 a numeric vector  
A2\_N03 a numeric vector  
A2\_P a numeric vector  
A2\_P205 a numeric vector  
A2\_Pb a numeric vector  
A2\_pH a numeric vector  
A2\_P04 a numeric vector  
A2\_Rb a numeric vector

A2\_S a numeric vector  
A2\_Sb a numeric vector  
A2\_Sb\_INAA a numeric vector  
A2\_Sc a numeric vector  
A2\_Sc\_INAA a numeric vector  
A2\_Se a numeric vector  
A2\_Se\_INAA a numeric vector  
A2\_Si a numeric vector  
A2\_SiO2 a numeric vector  
A2\_Sm\_INAA a numeric vector  
A2\_Sn\_INAA a numeric vector  
A2\_SO4 a numeric vector  
A2\_Sr a numeric vector  
A2\_Sr\_INAA a numeric vector  
A2\_Sum a numeric vector  
A2-Ta\_INAA a numeric vector  
A2\_Tb\_INAA a numeric vector  
A2\_Te a numeric vector  
A2\_Th a numeric vector  
A2\_Th\_INAA a numeric vector  
A2\_Ti a numeric vector  
A2\_TiO2 a numeric vector  
A2\_U\_INAA a numeric vector  
A2\_V a numeric vector  
A2\_W\_INAA a numeric vector  
A2\_Y a numeric vector  
A2\_Yb\_INAA a numeric vector  
A2\_Zn a numeric vector  
A2\_Zn\_INAA a numeric vector

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**Source**

Kola Project (1993-1998)

## References

Reimann C, Järvis M, Chekushin V, Bogatyrev I, Boyd R, Caritat P de, Dutter R, Finne TE, Halleraker JH, Jøger ?, Kashulina G, Lehto O, Niskavaara H, Pavlov V, Røisønen ML, Strand T, Volden T. Environmental Geochemical Atlas of the Central Barents Region. NGU-GTK-CKE Special Publication, Geological Survey of Norway, Trondheim, Norway, 1998.

## Examples

```
data(CHorANADUP)
str(CHorANADUP)
```

---

CHorFieldDUP

*Field duplicates of the C-horizon Kola data*

---

## Description

Field duplicates have been selected for quality control.

## Usage

```
data(CHorFieldDUP)
```

## Format

A data frame with 49 observations on the following 240 variables.

F1\_.Loc a numeric vector  
F2\_.Loc a numeric vector  
XC00 a numeric vector  
YC00 a numeric vector  
F1\_Ag a numeric vector  
F1\_Ag\_INAA a numeric vector  
F1\_Al a numeric vector  
F1\_Al203 a numeric vector  
F1\_As a numeric vector  
F1\_As\_INAA a numeric vector  
F1\_Au\_INAA a numeric vector  
F1\_B a numeric vector  
F1\_Ba a numeric vector  
F1\_Ba\_INAA a numeric vector  
F1\_Be a numeric vector  
F1\_Bi a numeric vector

F1\_Br a numeric vector  
F1\_Br\_INAA a numeric vector  
F1\_Ca a numeric vector  
F1\_Ca\_INAA a numeric vector  
F1\_CaO a numeric vector  
F1\_Cd a numeric vector  
F1\_Ce\_INAA a numeric vector  
F1\_Cl a numeric vector  
F1\_Co a numeric vector  
F1\_Co\_INAA a numeric vector  
F1\_Cond a numeric vector  
F1\_Cr a numeric vector  
F1\_Cr\_INAA a numeric vector  
F1\_Cs\_INAA a numeric vector  
F1\_Cu a numeric vector  
F1\_Eu\_INAA a numeric vector  
F1\_F a numeric vector  
F1\_F\_ionselect a numeric vector  
F1\_Fe a numeric vector  
F1\_Fe\_INAA a numeric vector  
F1\_Fe203 a numeric vector  
F1\_Hf\_INAA a numeric vector  
F1\_Hg a numeric vector  
F1\_Hg\_INAA a numeric vector  
F1\_Ir\_INAA a numeric vector  
F1\_K a numeric vector  
F1\_K20 a numeric vector  
F1\_La a numeric vector  
F1\_La\_INAA a numeric vector  
F1\_Li a numeric vector  
F1\_LOI a numeric vector  
F1\_Lu\_INAA a numeric vector  
F1\_Mass\_INAA a numeric vector  
F1\_Mg a numeric vector  
F1\_MgO a numeric vector  
F1\_Mn a numeric vector  
F1\_MnO a numeric vector



F1\_Mo a numeric vector  
F1\_Mo\_INAA a numeric vector  
F1\_Na a numeric vector  
F1\_Na\_INAA a numeric vector  
F1\_Na20 a numeric vector  
F1\_Nd\_INAA a numeric vector  
F1\_Ni a numeric vector  
F1\_Ni\_INAA a numeric vector  
F1\_N02 a numeric vector  
F1\_N03 a numeric vector  
F1\_P a numeric vector  
F1\_P205 a numeric vector  
F1\_Pb a numeric vector  
F1\_pH a numeric vector  
F1\_P04 a numeric vector  
F1\_Rb a numeric vector  
F1\_S a numeric vector  
F1\_Sb a numeric vector  
F1\_Sb\_INAA a numeric vector  
F1\_Sc a numeric vector  
F1\_Sc\_INAA a numeric vector  
F1\_Se a numeric vector  
F1\_Se\_INAA a numeric vector  
F1\_Si a numeric vector  
F1\_Si02 a numeric vector  
F1\_Sm\_INAA a numeric vector  
F1\_Sn\_INAA a numeric vector  
F1\_S04 a numeric vector  
F1\_Sr a numeric vector  
F1\_Sr\_INAA a numeric vector  
F1\_Sum a numeric vector  
F1-Ta\_INAA a numeric vector  
F1\_Tb\_INAA a numeric vector  
F1\_Te a numeric vector  
F1\_Th a numeric vector  
F1\_Th\_INAA a numeric vector  
F1\_Ti a numeric vector

F1\_TiO2 a numeric vector  
F1\_U\_INAA a numeric vector  
F1\_V a numeric vector  
F1\_W\_INAA a numeric vector  
F1\_Y a numeric vector  
F1\_Yb\_INAA a numeric vector  
F1\_Zn a numeric vector  
F1\_Zn\_INAA a numeric vector  
F2\_Ag a numeric vector  
F2\_Ag\_INAA a numeric vector  
F2\_Al a numeric vector  
F2\_Al2O3 a numeric vector  
F2\_As a numeric vector  
F2\_As\_INAA a numeric vector  
F2\_Au\_INAA a numeric vector  
F2\_B a numeric vector  
F2\_Ba a numeric vector  
F2\_Ba\_INAA a numeric vector  
F2\_Be a numeric vector  
F2\_Bi a numeric vector  
F2\_Br a numeric vector  
F2\_Br\_INAA a numeric vector  
F2\_Ca a numeric vector  
F2\_Ca\_INAA a numeric vector  
F2\_CaO a numeric vector  
F2\_Cd a numeric vector  
F2\_Ce\_INAA a numeric vector  
F2\_Cl a numeric vector  
F2\_Co a numeric vector  
F2\_Co\_INAA a numeric vector  
F2\_Cond a numeric vector  
F2\_Cr a numeric vector  
F2\_Cr\_INAA a numeric vector  
F2\_Cs\_INAA a numeric vector  
F2\_Cu a numeric vector  
F2\_Eu\_INAA a numeric vector  
F2\_F a numeric vector

F2\_F\_ionselect a numeric vector  
F2\_Fe a numeric vector  
F2\_Fe\_INAA a numeric vector  
F2\_Fe203 a numeric vector  
F2\_Hf\_INAA a numeric vector  
F2\_Hg a numeric vector  
F2\_Hg\_INAA a numeric vector  
F2\_Ir\_INAA a numeric vector  
F2\_K a numeric vector  
F2\_K20 a numeric vector  
F2\_La a numeric vector  
F2\_La\_INAA a numeric vector  
F2\_Li a numeric vector  
F2\_LOI a numeric vector  
F2\_Lu\_INAA a numeric vector  
F2\_Mass\_INAA a numeric vector  
F2\_Mg a numeric vector  
F2\_MgO a numeric vector  
F2\_Mn a numeric vector  
F2\_MnO a numeric vector  
F2\_Mo a numeric vector  
F2\_Mo\_INAA a numeric vector  
F2\_Na a numeric vector  
F2\_Na\_INAA a numeric vector  
F2\_Na20 a numeric vector  
F2\_Nd\_INAA a numeric vector  
F2\_Ni a numeric vector  
F2\_Ni\_INAA a numeric vector  
F2\_N02 a numeric vector  
F2\_N03 a numeric vector  
F2\_P a numeric vector  
F2\_P205 a numeric vector  
F2\_Pb a numeric vector  
F2\_pH a numeric vector  
F2\_P04 a numeric vector  
F2\_Rb a numeric vector  
F2\_S a numeric vector

F2\_Sb a numeric vector  
F2\_Sb\_INAA a numeric vector  
F2\_Sc a numeric vector  
F2\_Sc\_INAA a numeric vector  
F2\_Se a numeric vector  
F2\_Se\_INAA a numeric vector  
F2\_Si a numeric vector  
F2\_SiO2 a numeric vector  
F2\_Sm\_INAA a numeric vector  
F2\_Sn\_INAA a numeric vector  
F2\_SO4 a numeric vector  
F2\_Sr a numeric vector  
F2\_Sr\_INAA a numeric vector  
F2\_Sum a numeric vector  
F2-Ta\_INAA a numeric vector  
F2\_Tb\_INAA a numeric vector  
F2\_Te a numeric vector  
F2\_Th a numeric vector  
F2\_Th\_INAA a numeric vector  
F2\_Ti a numeric vector  
F2\_TiO2 a numeric vector  
F2\_U\_INAA a numeric vector  
F2\_V a numeric vector  
F2\_W\_INAA a numeric vector  
F2\_Y a numeric vector  
F2\_Yb\_INAA a numeric vector  
F2\_Zn a numeric vector  
F2\_Zn\_INAA a numeric vector  
DATE a numeric vector  
X.SAMP a factor with levels CRJHPC CRPCTF CRTF GKJHOJ GKJHTV JARR JHOJTV M?VG MLRJARP  
MLRJSRR MLRM?DR OJGKTV RPAV RPMLRJA RPVM Semenov Smirnov VGM?  
ELEV a numeric vector  
UTM a numeric vector  
X.COUN a factor with levels FIN NOR RUS  
X.ASP a factor with levels E FLAT N NE NW S SE SW  
X.GENLAN a factor with levels FLAT LOWMO PLAIN RIDGE SLOPE  
X.TOPO a factor with levels CONCLOW CONCMED CONVLOW CONVMED FLAT FLATLOW FLATTER LOWBROW  
LOWBRMED TER TERR TOP TOPFLAT TOPTER UPBRFLAT UPBROW UPBRMED UPBRTER

X.FORDEN a factor with levels D MD MD NO S

X.TREESPE a factor with levels BI BI . . BI .PBET .JUN BI . .PI .BI .SP BI . .SP BI .SP . BI .S .PJUN  
NO P P . P .BI P .BIJUN P .BI .S .PIBI . PI .BI PI . .BI PI .BI . .PIBI .SP PI . .SP PI . .SPBI  
P .SBI P .S .BI P .SBI .JUN S .BI S .BI .JUN SP . .BI SP .BI . .SPBI .PI .SPPIBI .

TRHIGH a numeric vector

RELAS a numeric vector

X.BUSHDEN a factor with levels MD NO S

X.BUSHSPEC a factor with levels BET BI . .BI .BI . BI . . .BI .JU BI . .JU BI . .PI JUN NO . .RO  
. .WI . .WIBI . .WIJU . .WIRO . .WIROJU

X.GRVEGETATIO a factor with levels B . .CGML B . .CH B .CO .GM B .CRCHMO .LIN B .CRGRMARMO .LI  
B .CRMO BJUO .MO .CR B .JUOMO .LI B .LINMAR B .MO .CRMAR .BO .ML C . . C . .BGML C .B .GML .C .BGML O  
C .B .GML O C .B .L C .BL .GM C .BM .HGL C .BML .GO C .BO .G C .BOM .L CH .BCRLIN CH .BLIN C .L .BGM  
C .M .GL C . .ML C .OL .M C .O .MLP CR .B .LI CR .LINMO H . .BML H .L .BCM L . .BMO L .BO .CM L .H .BM  
LIN .CR .LI M .BC .GL M . .BCL M .B .CLO M .BH .CGO M .B .L M .BL .GO M .O .BCGL MO .BCR MO .BCR JUO  
O .B .CHMLO

X.MOSS a factor with levels -9999 HSDC HSDR HSSC HSSR PS PSDC PSDR PSRD PSSC

X.TOP a factor with levels -9999 D10 D6 D7 M10 M4 M5 M6 M7 M8

AoMEAN a numeric vector

X.AoRANGE a factor with levels 0.1\_1.0 0\_2 0.2\_2.5 0.2\_4.0 0,5\_2 0,5\_3 0.5\_4.0 0.5\_5.0  
1.0\_3.0 1\_2 1\_3 1\_4 1\_5 1.5\_3.5 1,5\_5 1\_6 2\_2 2.0\_5.0 2.0\_6.0 2.0\_7.0 2\_3 2\_4 2\_5  
2\_6 2\_7 3.0\_8.0 3\_12 3\_5 3\_6 4\_12 4\_6 4\_8 5\_5\_10 .5\_4 -9999

HUMNO a numeric vector

HUMTHI a numeric vector

X.C\_PAR a factor with levels FLUV FLUVG TILL TILLSAP TILL&SAP

X.C\_grain a numeric vector

X.COLA a numeric vector

X.COLE a numeric vector

LOWDE a numeric vector

X.COLB a numeric vector

LOWDB a numeric vector

X.COLC a numeric vector

TOPC a numeric vector

X.WEATH a factor with levels DRY MIX RAIN

TEMP a numeric vector

CATLEV0 a numeric vector

CATLEV1 a numeric vector

CATLEV2 a numeric vector

LITO a numeric vector

F1\_Ag.1 a numeric vector

F1\_Ag.2 a numeric vector

F2\_Ag.1 a numeric vector  
 F1\_A1203.1 a numeric vector  
 F1\_A1203.2 a numeric vector  
 F2\_A1203.1 a numeric vector  
 F1\_Au\_INAA.1 a numeric vector  
 F1\_Au\_INAA.2 a numeric vector  
 F2\_Au\_INAA.1 a numeric vector  
 F1\_Ba\_INAA.1 a numeric vector  
 F1\_Ba\_INAA.2 a numeric vector  
 F2\_Ba\_INAA.1 a numeric vector

### Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

### Source

Kola Project (1993-1998)

### References

Reimann C, ?yr?s M, Chekushin V, Bogatyrev I, Boyd R, Caritat P de, Dutter R, Finne TE, Halleraker JH, J?ger ?, Kashulina G, Lehto O, Niskavaara H, Pavlov V, R?is?nen ML, Strand T, Volden T. Environmental Geochemical Atlas of the Central Barents Region. NGU-GTK-CKE Special Publication, Geological Survey of Norway, Trondheim, Norway, 1998.

### Examples

```
data(ChorFieldDUP)
str(ChorFieldDUP)
```

---

horizon

*C-horizon of the Kola Data*

---

### Description

The Kola Data were collected in the Kola Project (1993-1998, Geological Surveys of Finland (GTK) and Norway (NGU) and Central Kola Expedition (CKE), Russia). More than 600 samples in five different layers were analysed, this dataset contains the C-horizon.

### Usage

```
data(horizon)
```

**Format**

A data frame with 606 observations on the following 111 variables.

ID a numeric vector  
XCOO a numeric vector  
YCOO a numeric vector  
ELEV a numeric vector  
COUN a factor with levels FIN NOR RUS  
ASP a factor with levels E FLAT N NE NW NW S SE SW W  
TOPC a numeric vector  
LITO a numeric vector  
Ag a numeric vector  
Ag\_INAA a numeric vector  
Al a numeric vector  
Al\_XRF a numeric vector  
Al2O3 a numeric vector  
As a numeric vector  
As\_INAA a numeric vector  
Au a numeric vector  
Au\_INAA a numeric vector  
B a numeric vector  
Ba a numeric vector  
Ba\_INAA a numeric vector  
Be a numeric vector  
Bi a numeric vector  
Br\_IC a numeric vector  
Br\_INAA a numeric vector  
Ca a numeric vector  
Ca\_INAA a numeric vector  
Ca\_XRF a numeric vector  
CaO a numeric vector  
Cd a numeric vector  
Ce\_INAA a numeric vector  
Cl\_IC a numeric vector  
Co a numeric vector  
Co\_INAA a numeric vector  
Cr a numeric vector  
Cr\_INAA a numeric vector

Cs\_INAA a numeric vector  
Cu a numeric vector  
EC a numeric vector  
Eu\_INAA a numeric vector  
F\_IC a numeric vector  
Fe a numeric vector  
Fe\_INAA a numeric vector  
Fe\_XRF a numeric vector  
Fe203 a numeric vector  
Hf\_INAA a numeric vector  
Hg a numeric vector  
Hg\_INAA a numeric vector  
Ir\_INAA a numeric vector  
K a numeric vector  
K\_XRF a numeric vector  
K20 a numeric vector  
La a numeric vector  
La\_INAA a numeric vector  
Li a numeric vector  
LOI a numeric vector  
Lu\_INAA a numeric vector  
Mg a numeric vector  
Mg\_XRF a numeric vector  
MgO a numeric vector  
Mn a numeric vector  
Mn\_XRF a numeric vector  
MnO a numeric vector  
Mo a numeric vector  
Mo\_INAA a numeric vector  
Na a numeric vector  
Na\_INAA a numeric vector  
Na\_XRF a numeric vector  
Na20 a numeric vector  
Nd\_INAA a numeric vector  
Ni a numeric vector  
Ni\_INAA a numeric vector  
NO3\_IC a numeric vector



P a numeric vector  
P\_XRF a numeric vector  
P205 a numeric vector  
Pb a numeric vector  
Pd a numeric vector  
pH a numeric vector  
PO4\_IC a numeric vector  
Pt a numeric vector  
Rb a numeric vector  
S a numeric vector  
Sb a numeric vector  
Sb\_INAA a numeric vector  
Sc a numeric vector  
Sc\_INAA a numeric vector  
Se a numeric vector  
Se\_INAA a numeric vector  
Si a numeric vector  
Si\_XRF a numeric vector  
SiO2 a numeric vector  
Sm\_INAA a numeric vector  
Sn\_INAA a numeric vector  
SO4\_IC a numeric vector  
Sr a numeric vector  
Sr\_INAA a numeric vector  
Ta\_INAA a numeric vector  
Tb\_INAA a numeric vector  
Te a numeric vector  
Th a numeric vector  
Th\_INAA a numeric vector  
Ti a numeric vector  
Ti\_XRF a numeric vector  
TiO2 a numeric vector  
U\_INAA a numeric vector  
V a numeric vector  
W\_INAA a numeric vector  
Y a numeric vector  
Yb\_INAA a numeric vector  
Zn a numeric vector  
Zn\_INAA a numeric vector

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**Source**

Kola Project (1993-1998)

**References**

Reimann C, ?yr?s M, Chekushin V, Bogatyrev I, Boyd R, Caritat P de, Dutter R, Finne TE, Haller-aker JH, J?ger ?, Kashulina G, Lehto O, Niskavaara H, Pavlov V, R?is?nen ML, Strand T, Volden T. Environmental Geochemical Atlas of the Central Barents Region. NGU-GTK-CKE Special Publication, Geological Survey of Norway, Trondheim, Norway, 1998.

**Examples**

```
data(chorizon)
str(chorizon)
```

---

CHorSTANDARD

*Standard reference material for the Kola data*

---

**Description**

This is needed for quality control.

**Usage**

```
data(CHorSTANDARD)
```

**Format**

A data frame with 52 observations on the following 95 variables.

X.Loc a numeric vector

Ag a numeric vector

Ag\_INAA a numeric vector

Al a numeric vector

Al2O3 a numeric vector

As a numeric vector

As\_INAA a numeric vector

Au\_INAA a numeric vector

B a numeric vector

Ba a numeric vector

Ba\_INAA a numeric vector

Be a numeric vector  
Bi a numeric vector  
Br a numeric vector  
Br\_INAA a numeric vector  
Ca a numeric vector  
Ca\_INAA a numeric vector  
CaO a numeric vector  
Cd a numeric vector  
Ce\_INAA a numeric vector  
Cl. a numeric vector  
Co a numeric vector  
Co\_INAA a numeric vector  
Cond a numeric vector  
Cr a numeric vector  
Cr\_INAA a numeric vector  
Cs\_INAA a numeric vector  
Cu a numeric vector  
Eu\_INAA a numeric vector  
F. a numeric vector  
F\_ionselect a numeric vector  
Fe a numeric vector  
Fe\_INAA a numeric vector  
Fe203 a numeric vector  
Hf\_INAA a numeric vector  
Hg a numeric vector  
Hg\_INAA a numeric vector  
Ir\_INAA a numeric vector  
K a numeric vector  
K20 a numeric vector  
La a numeric vector  
La\_INAA a numeric vector  
Li a numeric vector  
LOI a numeric vector  
Lu\_INAA a numeric vector  
Mass\_INAA a numeric vector  
Mg a numeric vector  
MgO a numeric vector

Mn a numeric vector  
Mn0 a numeric vector  
Mo a numeric vector  
Mo\_INAA a numeric vector  
Na a numeric vector  
Na\_INAA a numeric vector  
Na20 a numeric vector  
Nd\_INAA a numeric vector  
Ni a numeric vector  
Ni\_INAA a numeric vector  
NO2. a numeric vector  
NO3. a numeric vector  
P a numeric vector  
P205 a numeric vector  
Pb a numeric vector  
pH a numeric vector  
PO4. . . a numeric vector  
Rb a numeric vector  
S a numeric vector  
Sb a numeric vector  
Sb\_INAA a numeric vector  
Sc a numeric vector  
Sc\_INAA a numeric vector  
Se a numeric vector  
Se\_INAA a numeric vector  
Si a numeric vector  
SiO2 a numeric vector  
Sm\_INAA a numeric vector  
Sn\_INAA a numeric vector  
SO4. . a numeric vector  
Sr a numeric vector  
Sr\_INAA a numeric vector  
Sum a numeric vector  
Ta\_INAA a numeric vector  
Tb\_INAA a numeric vector  
Te a numeric vector  
Th a numeric vector

Th\_INAA a numeric vector  
 Ti a numeric vector  
 TiO2 a numeric vector  
 U\_INAA a numeric vector  
 V a numeric vector  
 W\_INAA a numeric vector  
 Y a numeric vector  
 Yb\_INAA a numeric vector  
 Zn a numeric vector  
 Zn\_INAA a numeric vector

### Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

### Source

Kola Project (1993-1998)

### References

Reimann C, ?yr?s M, Chekushin V, Bogatyrev I, Boyd R, Caritat P de, Dutter R, Finne TE, Halleraker JH, J?ger ?, Kashulina G, Lehto O, Niskavaara H, Pavlov V, R?is?nen ML, Strand T, Volden T. Environmental Geochemical Atlas of the Central Barents Region. NGU-GTK-CKE Special Publication, Geological Survey of Norway, Trondheim, Norway, 1998.

### Examples

```
data(CHorSTANDARD)
str(CHorSTANDARD)
```

---

concarea

*Plot Concentration Area*

---

### Description

Displays a concentration-area plot (see also concareaExampleKola). This function is preferable since it can be applied to non-Kola data!

### Usage

```
concarea(x, y, z, zname = deparse(substitute(z)),
         caname = deparse(substitute(z)), borders=NULL, logx = FALSE, ifjit = FALSE,
         ifrev = FALSE, ngrid = 100, ncp = 0, xlim = NULL, xcoord = "Easting",
         ycoord = "Northing", ifbw = FALSE, x.logfinetick = c(2, 5, 10),
         y.logfinetick = c(2, 5, 10))
```

**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	a title for the x-axes of the qp-plot and concentration area plot.
caname	a title for the image of interpolated data.
borders	either NULL or character string with the name of the list with list elements x and y for x- and y-coordinates of map borders
logx	if it is required to make a logarithmic data transformation for the interpolation
ifrev	if FALSE the empirical function is plotted from highest value to lowest
ngrid	default value is 100
xlim	the range for the x-axis
xcoord	a title for the x-axis, defaults to "Easting"
ycoord	a title for the y-axis, defaults to "Northing"
ifbw	if the plot is drawn in black and white
x.logfinetick	how fine are the tick marks on log-scale on x-axis
y.logfinetick	how fine are the tick marks on log-scale on y-axis
ifjit	default value is FALSE
ncp	default value is 0

**Details**

The function assumes that the area is proportional to the count of grid points. To be a reasonable model the data points should be 'evenly' spread over the plane. The interpolated grid size is computed as  $(\max(x) - \min(x))/ngrid$ , with a default value of 100 for ngrid. Akima's interpolation function is used to obtain a linear interpolation between the spatial data values.

**Value**

The concentration area plot, in both directions, is created.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**See Also**

[concareaExampleKola](#)

**Examples**

```

data(ohorizon)
data(kola.background)
data(bordersKola)

Cu=ohorizon[, "Cu"]
X=ohorizon[, "XCO0"]
Y=ohorizon[, "YCO0"]

op <- par(mfrow=c(1,2),mar=c(4,4,2,2))
concarearea(X,Y,Cu,log=TRUE,zname="Cu in 0-horizon [mg/kg]",borders="bordersKola", ifrev=FALSE,
            x.logfinetick=c(2,5,10),y.logfinetick=c(10))
par(op)

```

---

concareareaExampleKola      *Concentration Area Plot for Kola data example*

---

**Description**

Displays a concentration area plot example for the Kola data. This procedure is useful for determining if multiple populations that are spatially dependent are present in a data set. For a more general function see concarearea.

**Usage**

```

concareareaExampleKola(x, y, z, zname = deparse(substitute(z)),
caname = deparse(substitute(z)), borders="bordersKola", logx = FALSE, ifjit = FALSE,
ifrev = FALSE, ngrid = 100, ncp = 0, xlim = NULL, xcoord = "Easting",
ycoord = "Northing", ifbw = FALSE, x.logfinetick = c(2, 5, 10),
y.logfinetick = c(2, 5, 10))

```

**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	a title for the x-axes of the qp-plot and concentration area plot.
caname	a title for the image of interpolated data.
borders	either NULL or character string with the name of the list with list elements x and y for x- and y-coordinates of map borders
logx	if it is required to make a logarithmic data transformation for the interpolation
ifrev	if FALSE the empirical function is plotted from highest value to lowest
ngrid	default value is 100
xlim	the range for the x-axis

xcoord	a title for the x-axis, defaults to "Easting"
ycoord	a title for the y-axis, defaults to "Northing"
ifbw	if the plot is drawn in black and white
x.logfinetick	how fine are the tick marks on log-scale on x-axis
y.logfinetick	how fine are the tick marks on log-scale on y-axis
ifjit	default value is FALSE
nep	default value is 0

### Details

The function assumes that the area is proportional to the count of grid points. To be a reasonable model the data points should be 'evenly' spread over the plane. The interpolated grid size is computed as  $(\max(x) - \min(x))/ngrid$ , with a default value of 100 for ngrid. Akima's interpolation function is used to obtain a linear interpolation between the spatial data values.

### Value

An example concentration area plot for Kola is created.

### Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

### References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

### See Also

[qqplot.das](#), [concareas](#)

### Examples

```
data(ohorizon)
data(kola.background)
data(bordersKola)

Cu=ohorizon[, "Cu"]
X=ohorizon[, "XCO0"]
Y=ohorizon[, "YCO0"]

op <- par(mfrow=c(2,2),mar=c(1.5,1.5,1.5,1.5))
concareasExampleKola(X,Y,Cu,log=TRUE,zname="Cu in 0-horizon [mg/kg]",
  x.logfinetick=c(2,5,10),y.logfinetick=c(10))
par(op)
```



---

`cor.sign`*Correlation Matrix*

---

**Description**

Computes correlation matrix of `x` with method "pearson", "kendall" or "spearman". This function also prints the matrix with the significance levels.

**Usage**

```
cor.sign(x, method = c("pearson", "kendall", "spearman"))
```

**Arguments**

<code>x</code>	the data
<code>method</code>	the method used

**Details**

This function estimate the association between paired samples and compute a test of the value being zero. All measures of association are in the range [-1,1] with 0 indicating no association.

**Value**

<code>cor</code>	Correlation matrix
<code>p.value</code>	p-value of the test statistic

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**See Also**

[cor.test](#)

**Examples**

```
data(chorizon)
x=chorizon[,c("Ca", "Cu", "Mg", "Na", "P", "Sr", "Zn")]

cor.sign(log10(x), method="spearman")
```

---

 CorCompare

*Compares Correlation Matrices*


---

**Description**

This function compares two correlation matrices numerically and graphically.

**Usage**

```
CorCompare(cor1, cor2, labels1, labels2, method1, method2, ndigits = 4,
lty1 = 1, lty2 = 2, col1 = 1, col2 = 2, lwd1 = 1.1, lwd2 = 1.1,
cex.label = 1.1, cex.legend = 1.2, lwd.legend = 1.2, cex.cor = 1, ...)
```

**Arguments**

cor1, cor2	two correlation matrices based on different estimation methods
labels1, labels2	labels for the two estimation methods
method1, method2	description of the estimation methods
ndigits	number of digits to be used for plotting the numbers
lty1, lty2, col1, col2, lwd1, lwd2, cex.label, cex.cor	other graphics parameters
cex.legend, lwd.legend	graphical parameters for the legend
...	further graphical parameters for the ellipses

**Details**

The ellipses are plotted with the function `do.ellipses`. Therefore the radius is calculated with singular value decomposition.

**Value**

No return value, creates a plot.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```

data(chorizon)
x=chorizon[,c("Ca", "Cu", "Mg", "Na", "P", "Sr", "Zn")]
op <- par(mfrow=c(1,1),mar=c(4,4,2,0))
R=robustbase::covMcd(log10(x),cor=TRUE)$cor
P=cor(log10(x))

CorCompare(R,P,labels1=dimnames(x)[[2]],labels2=dimnames(x)[[2]],
method1="Robust",method2="Pearson",ndigits=2, cex.label=1.2)
par(op)

```

CorGroups

*Correlation Matrix for Sub-groups***Description**

The correlation matrix for sub-groups of data is computed and displayed in a graphic.

**Usage**

```
CorGroups(dat, grouping, labels1, labels2, legend, ndigits = 4,
method = "pearson", ...)
```

**Arguments**

dat	data values (probably log10-transformed)
grouping	factor with levels for different groups
labels1, labels2	labels for groups
legend	plotting legend
ndigits	number of digits to be used for plotting the numbers
method	correlation method: "pearson", "spearman" or "kendall"
...	will not be used in the function

**Details**

The correlation is estimated with a non robust method but it is possible to select between the method of Pearson, Spearman and Kendall. The groups must be provided by the user.

**Value**

Graphic with the different sub-groups.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

## References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

## Examples

```
data(chorizon)
x=chorizon[,c("Ca","Cu","Mg","Na","P","Sr","Zn")]

#definition of the groups
lit=chorizon[,"LITO"]
litolog=rep(NA, length(lit))
litolog[lit=="10"] <- 1
litolog[lit=="52"] <- 2
litolog[lit=="81 | lit=="82 | lit=="83"] <- 3
litolog[lit=="7"] <- 4
litolog <- litolog[!is.na(litolog)]
litolog <- factor(litolog, labels=c("AB","PG","AR","LPS"))

op <- par(mfrow=c(1,1),mar=c(0.1,0.1,0.1,0.1))
CorGroups(log10(x), grouping=litolog, labels1=dimnames(x)[[2]],labels2=dimnames(x)[[2]],
legend=c("Caledonian Sediments","Basalts","Alkaline Rocks","Granites"),ndigits=2)
par(op)
```

---

do.ellipses

*Plot Ellipses*

---

## Description

This function plots ellipses according to a covariance matrix

## Usage

```
do.ellipses(acov, pos, ...)
```

## Arguments

acov	the given covariance matrix
pos	the location of the ellipse
...	further graphical parameter for the ellipses

## Details

The correlation matrix of the given covariance is computed and the resulting ellipse is plotted. The radi is computed with the singular value decomposition and the cos/sin is calculated for 100 different degrees.

**Value**

No return value, creates a plot.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
#internal function, used in CorCompare
```

---

edaplot	<i>EDA-plot for data</i>
---------	--------------------------

---

**Description**

This function plots a histogram of the data. There is also the choice to add the density, a boxplot and a scatterplot to the histogram.

**Usage**

```
edaplot(data,scatter=TRUE,box=TRUE, P.plot=TRUE, D.plot=TRUE,
        P.main=paste("Histogram of",deparse(substitute(data))),
        P.sub=NULL, P.xlab=deparse(substitute(data)), P.ylab=default, P.ann=par("ann"),
        P.axes=TRUE, P.frame.plot=P.axes, P.log=FALSE, P.logfine=c(2,5,10), P.xlim=NULL,
        P.cex.lab=1.4,B.range=1.5, B.notch=FALSE, B.outline=TRUE,
        B.border=par("fg"), B.col=NULL, B.pch=par("pch"), B.cex=1, B.bg=NA,
        H.breaks="Sturges", H.freq=TRUE, H.include.lowest=TRUE, H.right=TRUE,
        H.density=NULL, H.angle=45, H.col=NULL, H.border=NULL, H.labels=FALSE,
        S.pch=".", S.col=par("col"), S.bg=NA, S.cex=1, D.lwd=1,D.lty=1)
```

**Arguments**

data	data set
scatter	if TRUE the scatter plot is added
box	if TRUE a boxplot or boxplotlog is added
P.plot	if it is plotted or just a list is computed
D.plot	if TRUE the density is added
P.main, P.sub, P.xlab, P.ylab, P.ann	graphical parameters for the density, see plot

`P.axes`, `P.frame.plot`  
plots the y-axis with the ticker  
`P.log`  
if TRUE the x-axis is in log-scale  
`P.logfine`  
how fine the tickers are  
`P.xlim`, `P.cex.lab`  
further graphical parameters  
`B.range`, `B.notch`, `B.outline`, `B.border`, `B.col`, `B.pch`, `B.cex`, `B.bg`  
parameters for boxplot and boxplotlog function, see boxplot and boxplotlog  
`H.breaks`, `H.freq`, `H.include.lowest`, `H.right`, `H.density`, `H.angle`, `H.col`,  
`H.border`, `H.labels`  
parameters for histogram, see hist  
`S.pch`, `S.col`, `S.bg`, `S.cex`  
graphical parameters for the shape of the points, see points  
`D.lwd`, `D.lty`  
parameters for the density

### Details

First the histogram, boxplot/boxplotlog and density is calculate and then the plot is produced. The default is that histogram, boxplot, density trace and scatterplot is made.

### Value

`H`  
results of the histogram  
`B`  
results of the boxplot

### Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

### References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

### See Also

[plot](#), [boxplot](#), [edaplotlog](#), [hist](#), [points](#)

### Examples

```

data(chorizon)
Ba=chorizon[, "Ba"]
edaplot(Ba,H.freq=FALSE,box=TRUE,H.breaks=30,S.pch=3,S.cex=0.5,D.lwd=1.5,P.log=FALSE,
P.main="",P.xlab="Ba [mg/kg]",P.ylab="Density",B.pch=3,B.cex=0.5)

```

---

edaplotlog

*Edaplot for logtransformed data*


---

## Description

This function plots a histogram of the data. There is also the choice to add the density, a boxplot and a scatterplot to the histogram.

## Usage

```
edaplotlog(data, scatter = TRUE, box = TRUE, P.plot = TRUE, D.plot = TRUE,
P.main = paste("Histogram of", deparse(substitute(data))), P.sub = NULL,
P.xlab = deparse(substitute(data)), P.ylab = default, P.ann = par("ann"),
P.axes = TRUE, P.frame.plot = P.axes, P.log = FALSE,
P.logfine = c(2, 5, 10), P.xlim = NULL, P.cex.lab = 1.4, B.range = 1.5,
B.notch = FALSE, B.outline = TRUE, B.border = par("fg"), B.col = NULL,
B.pch = par("pch"), B.cex = 1, B.bg = NA, B.log = FALSE,
H.breaks = "Sturges", H.freq = TRUE, H.include.lowest = TRUE,
H.right = TRUE, H.density = NULL, H.angle = 45, H.col = NULL,
H.border = NULL, H.labels = FALSE, S.pch = ".", S.col = par("col"),
S.bg = NA, S.cex = 1, D.lwd = 1, D.lty = 1)
```

## Arguments

data	data set
scatter	if TRUE the scatter plot is added
box	if TRUE a boxplot or boxplotlog is added
P.plot	if it is plotted or just a list is computed
D.plot	if TRUE the density is added
P.main, P.sub, P.xlab, P.ylab, P.ann	graphical parameters for the density, see plot
P.axes, P.frame.plot	plots the y-axis with the ticker
P.log	if TRUE the x-axis is in log-scale
P.logfine	how fine the tickers are
P.xlim, P.cex.lab	further graphical parameters
B.range, B.notch, B.outline, B.border, B.col, B.pch, B.cex, B.bg	parameters for boxplot and boxplotlog function, see boxplot and boxplotlog
B.log	if TRUE the function boxplotlog is used instead of boxplot
H.breaks, H.include.lowest, H.right, H.density, H.angle, H.col, H.border, H.labels	parameters for histogram, see hist

H.freq uses the number of data points in the range  
 S.pch, S.col, S.bg, S.cex graphical parameters for the shape of the points, see points  
 D.lwd, D.lty parameters for the density

### Details

First the histogram, boxplot/boxplotlog and density is calculate and then the plot is produced. The default is that histogram, boxplot, density trace and scatterplot is made.

### Value

H results of the histogram  
 B results of boxplotlog

### Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

### References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

### See Also

[plot](#), [boxplot](#), [boxplotlog](#), [hist](#), [points](#)

### Examples

```
data(chorizon)
Ba=chorizon[, "Ba"]
edaplotlog(Ba,H.freq=FALSE,box=TRUE,H.breaks=30,S.pch=3,S.cex=0.5,D.lwd=1.5,P.log=FALSE,
  P.main="",P.xlab="Ba [mg/kg]",P.ylab="Density",B.pch=3,B.cex=0.5,B.log=TRUE)
```

---

factanal.fit.principal

*Fit a Factor Analysis*

---

### Description

Internal function for pfa.

### Usage

```
factanal.fit.principal(cmat, factors, p = ncol(cmat), start = NULL,
  iter.max = 10, unique.tol = 1e-04)
```



**Arguments**

cmat	provided correlation matrix
factors	number of factors
p	number of observations
start	vector of start values
iter.max	maximum number of iteration used to calculate the common factor
unique.tol	the tolerance for a deviation of the maximum (in each row, without the diag) value of the given correlation matrix to the new calculated value

**Value**

loadings	A matrix of loadings, one column for each factor. The factors are ordered in decreasing order of sums of squares of loadings.
uniqueness	uniqueness
correlation	correlation matrix
criteria	The results of the optimization: the value of the negativ log-likelihood and information of the iterations used.
factors	the factors
dof	degrees of freedom
method	"principal"

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

---

kola.background      *kola.background*

---

**Description**

Coordinates of the Kola background. Seperate polygons for the project boundary, borders, lakes and coast are provided.

**Usage**

data(kola.background)

**Format**

The format is: List of 4 \$ boundary:'data.frame': 50 obs. of 2 variables: ..\$ V1: num [1:50] 388650 388160 386587 384035 383029 ... ..\$ V2: num [1:50] 7892400 7881248 7847303 7790797 7769214 ... \$ coast :'data.frame': 6259 obs. of 2 variables: ..\$ V1: num [1:6259] 438431 439102 439102 439643 439643 ... ..\$ V2: num [1:6259] 7895619 7896495 7896495 7895800 7895542 ... \$ borders :'data.frame': 504 obs. of 2 variables: ..\$ V1: num [1:504] 417575 417704 418890 420308 422731 ... ..\$ V2: num [1:504] 7612984 7612984 7613293 7614530 7615972 ... \$ lakes :'data.frame': 6003 obs. of 2 variables: ..\$ V1: num [1:6003] 547972 546915 NA 547972 547172 ... ..\$ V2: num [1:6003] 7815109 7815599 NA 7815109 7813873 ...

**Details**

Is used by plotbg()

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**Source**

Kola Project (1993-1998)

**References**

Reimann C, Ayras M, Chekushin V, Bogatyrev I, Boyd R, Caritat P de, Dutter R, Finne TE, Halleraker JH, Jager O, Kashulina G, Lehto O, Niskavaara H, Pavlov V, Raisanen ML, Strand T, Volden T. Environmental Geochemical Atlas of the Central Barents Region. NGU-GTK-CKE Special Publication, Geological Survey of Norway, Trondheim, Norway, 1998.

**Examples**

```
data(kola.background)
plotbg()
```

---

KrigeLegend

*Krige*

---

**Description**

Plots Krige maps and Legend based on continuous or percentile scale.

**Usage**

```
KrigeLegend(X, Y, z, resol = 100, vario, type = "percentile",
whichcol = "gray", qtiles = c(0, 0.05, 0.25, 0.5, 0.75, 0.9, 0.95, 1), borders=NULL,
leg.xpos.min = 780000, leg.xpos.max = 8e+05, leg.ypos.min = 7760000,
leg.ypos.max = 7870000, leg.title = "mg/kg", leg.title.cex = 0.7,
leg.numb.cex = 0.7, leg.round = 2, leg.numb.xshift = 70000, leg.perc.xshift = 40000,
leg.perc.yshift = 20000, tit.xshift = 35000)
```

**Arguments**

X	X-coordinates
Y	Y-coordinates
z	values on the coordinates
resol	resolution of blocks for Kriging
vario	variogram model
type	"percentile" for percentile legend, "contin" for continuous grey-scale or colour map
whichcol	type of colour scheme to use: "gray", "rainbow", "rainbow.trunc", "rainbow.inv", "terrain", "topo"
qtiles	considered quantiles if type="percentile" is used
borders	either NULL or character string with the name of the list with list elements x and y for x- and y-coordinates of map borders
leg.xpos.min	minimum value of x-position of the legend
leg.xpos.max	maximum value of x-position of the legend
leg.ypos.min	minimum value of y-position of the legend
leg.ypos.max	maximum value of y-position of the legend
leg.title	title for legend
leg.title.cex	cex for legend title
leg.numb.cex	cex for legend number
leg.round	round legend to specified digits "pretty"
leg.numb.xshift	x-shift of numbers in legend relative to leg.xpos.max
leg.perc.xshift	x-shift of "Percentile" in legend relative to leg.xpos.min
leg.perc.yshift	y-shift of numbers in legend relative to leg.ypos.max
tit.xshift	x-shift of title in legend relative to leg.xpos.max

**Details**

Based on a variogram model a interpolation of the spatial data is computed. The variogram has to be provided by the user and based on this model the spatial prediction is made. To distinguish between different values every predicted value is plotted in his own scale of the choosen colour.

**Value**

No return value, creates a plot.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

## References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

## Examples

```
data(chorizon)
data(kola.background)
X=chorizon[,"XC00"]
Y=chorizon[,"YC00"]
#el=chorizon[,"As"]
#vario.b <- variog(coords=cbind(X,Y), data=el, lambda=0, max.dist=300000)
#data(res.eyefit.As_C_m) #need the data
#v5=variofit(vario.b,res.eyefit.As_C_m,cov.model="spherical",max.dist=300000)

plot(X,Y,frame.plot=FALSE,xaxt="n",yaxt="n",xlab="",ylab="",type="n")

# to increase the resolution, set e.g. resol=100
#data(bordersKola) # x and y coordinates of project boundary
#KrigeLegend(X,Y,el,resol=25,vario=v5,type="percentile",whichcol="gray",
#  qtiles=c(0,0.05,0.25,0.50,0.75,0.90,0.95,1),borders="bordersKola",
#  leg.xpos.min=7.8e5,leg.xpos.max=8.0e5,leg.ypos.min=77.6e5,leg.ypos.max=78.7e5,
#  leg.title="mg/kg", leg.title.cex=0.7, leg.numb.cex=0.7, leg.round=2,
#  leg.numb.xshift=0.7e5,leg.perc.xshift=0.4e5,leg.perc.yshift=0.2e5,tit.xshift=0.35e5)
#
#plotbg(map.col=c("gray","gray","gray","gray"),map.lwd=c(1,1,1,1),add.plot=TRUE)
```

---

loadplot

*Plot the Loadings of a FA*

---

## Description

Makes a Reimann-plot of a loadings matrix.

## Usage

```
loadplot(fa.object, titlepl = "Factor Analysis", crit = 0.3, length.varnames = 2)
```

## Arguments

fa.object	the output of factor analysis class
titlepl	the title of the plot
crit	all loadings smaller than crit will be ignored in the plot
length.varnames	number of letters for abbreviating the variable names in the plot

## Value

Plot of the loadings of a FA therefore a object of factor analysis class must be provided.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
data(moss)
var=c("Ag", "Al", "As", "B", "Ba", "Bi", "Ca", "Cd", "Co", "Cr", "Cu", "Fe", "Hg", "K", "Mg", "Mn", "Mo",
      "Na", "Ni", "P", "Pb", "Rb", "S", "Sb", "Si", "Sr", "Th", "Tl", "U", "V", "Zn")
x=log10(moss[,var])

x.mcd=robustbase::covMcd(x, cor=TRUE)
x.rsc=scale(x, x.mcd$cent, sqrt(diag(x.mcd$cov)))
res5=pfa(x.rsc, factors=2, covmat=x.mcd, scores="regression", rotation="varimax",
        maxit=0, start=rep(0, ncol(x.rsc)))
loadplot(res5, title1="Robust FA (log-transformed)", crit=0.3)
```

---

monch

*Boundary of the Monchegorsk area*

---

**Description**

This gives x- and y-coordinates with the boundary of the area around Monchegorsk.

**Usage**

```
data(monch)
```

**Format**

The format is: List of 2 \$ x: num [1:32] 710957 734664 754666 770223 779113 ... \$ y: num [1:32] 7473981 7473143 7474818 7483191 7488215 ...

**Details**

This object can be used to select samples from the Kola data from the region around Monchegorsk.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

## References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

## Examples

```
data(monch)
data(kola.background)
plotbg()
lines(monch$x,monch$y,col="red")
```

---

moss

*Moss layer of the Kola Data*

---

## Description

The Kola Data were collected in the Kola Project (1993-1998, Geological Surveys of Finland (GTK) and Norway (NGU) and Central Kola Expedition (CKE), Russia). More than 600 samples in five different layers were analysed, this dataset contains the moss layer.

## Usage

```
data(moss)
```

## Format

A data frame with 594 observations on the following 58 variables.

ID a numeric vector

XCOO a numeric vector

YCOO a numeric vector

ELEV a numeric vector

COUN a factor with levels FIN NOR RUS

ASP a factor with levels E FLAT N NE NW NW S SE SW W

GENLAN a factor with levels DEEPVAL FLA PLAIN FLAT HIMO LOWMO PLAIN PLAT RIDGE SLOPE

TOPO a factor with levels BRUP BRUPLow BRUPSTEE CONC CONCFLAT CONCLow CONCmed CONCRUG  
CONCTERR CONV CONVLO CONVLOW CONVmed CONVTER FLAT FLATLOW FLATRUG FLATTER FLATTERR  
LOBRRUG LOW LOWBR LOWBRFLAT LOWBRLO LOWBRLOW LOWBRMED RUG RUGLOW TER TERLOW TERR  
TERRLOW TOHIFLAT TOP TOPFLAT TOPHILO TOPLOW TOPTER TOPUPBR UPBR UPBRFLAT UPBRLow  
UPBRMED UPBRTER UPBRTERR UPTER

GROUNDVEG a factor with levels BLUEBERRY CARLIN\_HEATHER EMPETRUM GRASS LICHEN MOSS SHRUBS  
WHITE\_LICHEN

TREELAY a factor with levels BIPI BIPIspr BIRCH BIRChdense BISPR BISPRPI MIX PIBI PIBISPR  
PINE PISPR PISPRBI SHRUBS SPARCEBI SPARCEPI SPRBI SPRBIPI SPRPI SPRPIBI SPRUCE  
WILLOW

VEG\_ZONE a factor with levels BOREAL\_FOREST DWARF\_SHRUB\_TUNDRA FOREST\_TUNDRA SHRUB\_TUNDRA TUNDRA

Date a numeric vector

SAMP a factor with levels ALL ATMLRMA CRGKPCF CRJHOJTV CRJHPC CRJHTF CROJTV CRPCTF CRPCTV CRTF DRMLRKK DRMRLKK GKJHOJ GKJHTV GKOJPCF GKOJTF GKOJTV GKPCTF HARR JA JAMAMRL JAMLRMA JAMLRMR JARKP JARP JARPMA JARPMLR JARR JARRMLR JCPCTF JHGKTV JHOJGK JHOJTV JHPCTF JHRBTV Katanaev MAKKVG MARP MARPMLR MARPMRL MAVG MLR MLRJA MLRJARP MLRJARR MLRJSRR MLRMADR MLRMAJA MLRMARP MLRMAVG MLRM?VG MLRRPJA MLRRPMA MRLMAJA OJGKTV OJTF Pavlov RPAV RPEM RPMA RPMLRJA RPMLRMA RPVM Semenov Smirnov TFOJ VGHNMA VGMA VGMAHN VGMARS VGMASR VGRSMA VMRP VMRPMA

SPECIES a factor with levels HSDC HSDR HSRC HSSC HSSR PS PSDC PSDR PSRC PSRD PSSC PSSR SFDR

LITO a numeric vector

C\_PAR a factor with levels BEDR FLUV FLUVG MAR SAP SEA STRAT TILL TILLSA TILLSAP TILL&SAP

TOPC a numeric vector

WEATH a factor with levels DRY DRY MIX MIX RAIN SNOW

TEMP a numeric vector

Ag a numeric vector

Al a numeric vector

As a numeric vector

Au a numeric vector

B a numeric vector

Ba a numeric vector

Be a numeric vector

Bi a numeric vector

Ca a numeric vector

Cd a numeric vector

Co a numeric vector

Cr a numeric vector

Cu a numeric vector

Fe a numeric vector

Hg a numeric vector

K a numeric vector

La a numeric vector

Mg a numeric vector

Mn a numeric vector

Mo a numeric vector

Na a numeric vector

Ni a numeric vector

P a numeric vector

Pb a numeric vector  
Pd a numeric vector  
Pt a numeric vector  
Rb a numeric vector  
S a numeric vector  
Sb a numeric vector  
Sc a numeric vector  
Se a numeric vector  
Si a numeric vector  
Sr a numeric vector  
Th a numeric vector  
Tl a numeric vector  
U a numeric vector  
V a numeric vector  
Y a numeric vector  
Zn a numeric vector

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**Source**

Kola Project (1993-1998)

**References**

Reimann C, ?yr?s M, Chekushin V, Bogatyrev I, Boyd R, Caritat P de, Dutter R, Finne TE, Halleraker JH, J?ger ?, Kashulina G, Lehto O, Niskavaara H, Pavlov V, R?is?nen ML, Strand T, Volden T. Environmental Geochemical Atlas of the Central Barents Region. NGU-GTK-CKE Special Publication, Geological Survey of Norway, Trondheim, Norway, 1998.

**Examples**

```
data(moss)
str(moss)
```



---

`nizap`*Boundary of the area Nickel-Zapoljarnij*

---

**Description**

This gives x- and y-coordinates with the boundary of the area around Nickel-Zapoljarnij.

**Usage**

```
data(nizap)
```

**Format**

The format is: List of 2 \$ x: num [1:36] 699104 693918 681324 662062 645023 ... \$ y: num [1:36] 7739416 7746115 7751139 7756163 7757000 ...

**Details**

This object can be used to select samples from the Kola data from the region around Nickel-Zapoljarnij.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
data(nizap)
data(kola.background)
plotbg()
lines(nizap$x, nizap$y, col="red")
```

---

`Northarrow`*Northarrow*

---

**Description**

Add a North Arrow to a map.

**Usage**

```
Northarrow(Xbottom, Ybottom, Xtop, Ytop, Xtext, Ytext, Alength, Aangle, Alwd,
Tcex)
```

**Arguments**

Xbottom	x coordinate of the first point
Ybottom	y coordinate of the first point
Xtop	x coordinate of the second point
Ytop	y coordinate of the second point
Xtext	x coordinate of the label
Ytext	y coordinate of the label
Alength	length of the edges of the arrow head (in inches)
Aangle	angle from the shaft of the arrow to the edge of the arrow head
Alwd	The line width, a positive number
Tcex	numeric character expansion factor

**Value**

No return value, creates a plot.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
plot.new()  
Northarrow(0.5,0,0.5,1,0.5,0.5,Alength=0.15,Aangle=15,Alwd=2,Tcex=2)
```

---

ohorizon

*O-horizon of the Kola Data*

---

**Description**

The Kola Data were collected in the Kola Project (1993-1998, Geological Surveys of Finland (GTK) and Norway (NGU) and Central Kola Expedition (CKE), Russia). More than 600 samples in five different layers were analysed, this dataset contains the O-horizon.

**Usage**

```
data(ohorizon)
```

**Format**

A data frame with 617 observations on the following 85 variables.

ID a numeric vector

XCOO a numeric vector

YCOO a numeric vector

ELEV a numeric vector

COUN a factor with levels FIN NOR RUS

X.ASP a factor with levels -9999 E FLAT N NE NW NW S SE SW W

AoMEAN a numeric vector

HUMNO a numeric vector

HUMTHI a numeric vector

GROUNDVEG a factor with levels BLUEBERRY CARLIN\_HEATHER EMPETRUM GRASS LICHEN MOSS SHRUBS  
WHITE\_LICHEN

TREELAY a factor with levels BIPI BIPISPR BIRCH BIRCHdense BISPR BISPRPI MIX PIBI PIBISPR  
PINE PISPR PISPRBI SHRUBS SPARCEBI SPARCEPI SPRBI SPRBIPI SPRPI SPRPIBI SPRUCE  
WILLOW

VEG\_ZONE a factor with levels BOREAL\_FOREST DWARF\_SHRUB\_TUNDRA FOREST\_TUNDRA SHRUB\_TUNDRA  
TUNDRA

LITO a numeric vector

Ag a numeric vector

A1 a numeric vector

A1\_AA a numeric vector

As a numeric vector

Au a numeric vector

B a numeric vector

Ba a numeric vector

Ba\_AA a numeric vector

Be a numeric vector

Bi a numeric vector

Br a numeric vector

C a numeric vector

Ca a numeric vector

Ca\_AA a numeric vector

Cd a numeric vector

Cd\_AA a numeric vector

C1 a numeric vector

Co a numeric vector

Co\_AA a numeric vector

Cond a numeric vector  
Cr a numeric vector  
Cr\_AA a numeric vector  
Cu a numeric vector  
Cu\_AA a numeric vector  
F a numeric vector  
Fe a numeric vector  
Fe\_AA a numeric vector  
H a numeric vector  
Hg a numeric vector  
K a numeric vector  
K\_AA a numeric vector  
La a numeric vector  
LOI a numeric vector  
Mg a numeric vector  
Mg\_AA a numeric vector  
Mn a numeric vector  
Mn\_AA a numeric vector  
Mo a numeric vector  
N a numeric vector  
Na a numeric vector  
Na\_AA a numeric vector  
Ni a numeric vector  
Ni\_AA a numeric vector  
NO3 a numeric vector  
P a numeric vector  
P\_AA a numeric vector  
Pb a numeric vector  
Pb\_AA a numeric vector  
Pd a numeric vector  
pH a numeric vector  
PO4 a numeric vector  
Pt a numeric vector  
Rb a numeric vector  
S a numeric vector  
S\_AA a numeric vector  
Sb a numeric vector

Sc a numeric vector  
Se a numeric vector  
Si a numeric vector  
Si\_AA a numeric vector  
S04 a numeric vector  
Sr a numeric vector  
Sr\_AA a numeric vector  
Th a numeric vector  
Ti\_AA a numeric vector  
Tl a numeric vector  
U a numeric vector  
V a numeric vector  
V\_AA a numeric vector  
Y a numeric vector  
Zn a numeric vector  
Zn\_AA a numeric vector

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**Source**

Kola Project (1993-1998)

**References**

Reimann C, ?yr?s M, Chekushin V, Bogatyrev I, Boyd R, Caritat P de, Dutter R, Finne TE, Halleraker JH, J?ger ?, Kashulina G, Lehto O, Niskavaara H, Pavlov V, R?is?nen ML, Strand T, Volden T. Environmental Geochemical Atlas of the Central Barents Region. NGU-GTK-CKE Special Publication, Geological Survey of Norway, Trondheim, Norway, 1998.

**Examples**

```
data(ohorizon)
str(ohorizon)
```

**Description**

Computes the principal factor analysis of the input data.

**Usage**

```
pfa(x, factors, data = NULL, covmat = NULL, n.obs = NA, subset, na.action,
    start = NULL, scores = c("none", "regression", "Bartlett"),
    rotation = "varimax", maxiter = 5, control = NULL, ...)
```

**Arguments**

x	(robustly) scaled input data
factors	number of factors
data	default value is NULL
covmat	(robustly) computed covariance or correlation matrix
n.obs	number of observations
subset	if a subset is used
start	starting values
scores	which method should be used to calculate the scores
rotation	if a rotation should be made
maxiter	maximum number of iterations
control	default value is NULL
na.action	what to do with NA values
...	arguments for creating a list

**Value**

loadings	A matrix of loadings, one column for each factor. The factors are ordered in decreasing order of sums of squares of loadings.
uniqueness	uniqueness
correlation	correlation matrix
criteria	The results of the optimization: the value of the negativ log-likelihood and information of the iterations used.
factors	the factors
dof	degrees of freedom
method	"principal"
n.obs	number of observations if available, or NA
call	The matched call.
STATISTIC, PVAL	The significance-test statistic and p-value, if can be computed

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
data(moss)
var=c("Ni", "Cu", "Mg", "Rb", "Mn")
x=log10(moss[,var])

x.mcd=robustbase::covMcd(x,cor=TRUE)
x.rsc=scale(x,x.mcd$cent,sqrt(diag(x.mcd$cov)))
pfa(x.rsc,factors=2,covmat=x.mcd,scores="regression",rotation="varimax",
    maxit=0,start=rep(0,ncol(x.rsc)))
```

---

plotbg

*Kola background Plot*

---

**Description**

Plots the Kola background

**Usage**

```
plotbg(map = "kola.background", which.map = c(1, 2, 3, 4),
map.col = c(5, 1, 3, 4), map.lwd = c(2, 1, 2, 1), add.plot = FALSE, ...)
```

**Arguments**

map	List of coordinates. For the correct format see also <code>help(kola.background)</code>
which.map	which==1 ... plot project boundary; which==2 ... plot coast line; which==3 ... plot country borders; which==4 ... plot lakes and rivers
map.col	Map colors to be used
map.lwd	Defines linestyle of the background
add.plot	logical. if true background is added to an existing plot
...	additional plot parameters, see <code>help(par)</code>

**Details**

Plots the background map of Kola

**Value**

No return value, creates a plot.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

Reimann C, ?yr?s M, Chekushin V, Bogatyrev I, Boyd R, Caritat P de, Dutter R, Finne TE, Halleraker JH, J?ger ?, Kashulina G, Lehto O, Niskavaara H, Pavlov V, R?is?nen ML, Strand T, Volden T. Environmental Geochemical Atlas of the Central Barents Region. NGU-GTK-CKE Special Publication, Geological Survey of Norway, Trondheim, Norway, 1998.

**Examples**

```
data(kola.background)
plotbg()
```

---

plotelement

*Plot Elements of a Discriminant Analysis*

---

**Description**

Plot the elements for the discriminant analysis. The plot is ordered in the different groups.

**Usage**

```
plotelement(da.object)
```

**Arguments**

da.object      a object of the lda class

**Value**

No return value, creates a plot.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.



**Examples**

```
data(iris3)
Iris <- data.frame(rbind(iris3[, ,1], iris3[, ,2], iris3[, ,3]), Sp = rep(c("s", "c", "v"), rep(50,3)))
train <- sample(1:150, 75)
z <- MASS::lda(Sp ~ ., Iris, prior = c(1,1,1)/3, subset = train)

plotelement(z)
```

---

plotellipse

*Plot Ellipse*

---

**Description**

Plots an ellipse with percentage tolerance and a certain location and covariance.

**Usage**

```
plotellipse(x.loc, x.cov, perc = 0.98, col = NULL, lty = NULL)
```

**Arguments**

x.loc	the location vector
x.cov	the covariance
perc	defines the percentage and should be a (vector of) number(s) between 0 and 1
col, lty	graphical parameters

**Details**

First the radius of the covariance is calculated and then the ellipses for the provided percentages are plotted at the certain location.

**Value**

Plot with ellipse.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```

data(moss)
Ba=log10(moss[, "Ba"])
Ca=log10(moss[, "Ca"])
plot.new()
plot.window(xlim=range(Ba),ylim=c(min(Ca)-1,max(Ca)))

x=cbind(Ba,Ca)
plotellipse(apply(x,2,mean),cov(x),perc=c(0.5,0.75,0.9,0.98))

```

---

plotmvoutlier

*Multivariate outlier plot*


---

**Description**

This function plots multivariate outliers. One possibility is to distinguish between outlier and no outlier. The alternative is to distinguish between the different percentils (e.g. <25%, 25%<x<50%,...).

**Usage**

```

plotmvoutlier(coord, data, quan = 1/2, alpha = 0.025, symb = FALSE, bw = FALSE,
plotmap = TRUE, map = "kola.background", which.map = c(1, 2, 3, 4),
map.col = c(5, 1, 3, 4), map.lwd = c(2, 1, 2, 1), pch2 = c(3, 21),
cex2 = c(0.7, 0.2), col2 = c(1, 1), lcex.fac = 1, ...)

```

**Arguments**

coord	the coordinates for the points
data	the value for the different coordinates
quan	Number of subsets used for the robust estimation of the covariance matrix. Allowed are values between 0.5 and 1., see covMcd
alpha	Maximum thresholding proportion
symb	if FALSE, only two different symbols (outlier and no outlier) will be used
bw	if TRUE, symbols are in gray-scale (only if symb=TRUE)
plotmap	if TRUE, the map is plotted
map	the name of the background map
which.map, map.col, map.lwd	parameters for the background plot, see plotbg
pch2, cex2, col2	graphical parameters for the points
lcex.fac	factor for multiplication of symbol size (only if symb=TRUE)
...	further parameters for the plot

**Details**

The function computes a robust estimation of the covariance and then the Mahalanobis distances are calculated. With this distances the data set is divided into outliers and non outliers. If `symb=FALSE` only two different symbols are used otherwise different grey scales are used to distinguish the different types of outliers.

**Value**

<code>o</code>	returns the outliers
<code>md</code>	the square root of the Mahalanobis distance
<code>euclidean</code>	the Euclidean distance of the scaled data

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**See Also**

[plotbg](#), [covMcd](#), [arw](#)

**Examples**

```
data(moss)
X=moss[, "XC00"]
Y=moss[, "YC00"]
el=c("Ag", "As", "Bi", "Cd", "Co", "Cu", "Ni")
x=log10(moss[, el])

data(kola.background)
plotmvoutlier(cbind(X,Y), x, symb=FALSE, map.col=c("grey", "grey", "grey", "grey"),
              map.lwd=c(1,1,1,1),
              xlab="", ylab="", frame.plot=FALSE, xaxt="n", yaxt="n")
```

---

plotuniout

*Multivariate outlier plot for each dimension*

---

**Description**

A multivariate outlier plot for each dimension is produced.

**Usage**

```
plotuniout(x, symb = FALSE, quan = 1/2, alpha = 0.025, bw = FALSE,
           pch2 = c(3, 1), cex2 = c(0.7, 0.4), col2 = c(1, 1), lcex.fac = 1, ...)
```

**Arguments**

x	dataset
symb	if FALSE, only two different symbols (outlier and no outlier) will be used
quan	Number of subsets used for the robust estimation of the covariance matrix. Allowed are values between 0.5 and 1., see covMcd
alpha	Maximum thresholding proportion, see arw
bw	if TRUE, symbols are in gray-scale (only if symb=TRUE)
pch2, cex2, col2	graphical parameters for the points
lcex.fac	factor for multiplication of symbol size (only if symb=TRUE)
...	further graphical parameters for the plot

**Value**

o	returns the outliers
md	the square root of the Mahalanobis distance
euclidean	the Euclidean distance of the scaled data

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**See Also**

[arw](#), [covMcd](#)

**Examples**

```
data(moss)
e1=c("Ag", "As", "Bi", "Cd", "Co", "Cu", "Ni")
dat=log10(moss[,e1])

ans<-plotuniout(dat, symb=FALSE, cex2=c(0.9, 0.1), pch2=c(3, 21))
```

---

polygrid	<i>Coordinates of Points Inside a Polygon</i>
----------	---

---

### Description

This function builds a rectangular grid and extracts points which are inside of an internal polygonal region.

### Usage

```
polygrid(xgrid, ygrid, borders, vec.inout = FALSE, ...)
```

### Arguments

xgrid	grid values in the $x$ -direction.
ygrid	grid values in the $y$ -direction.
borders	a matrix with polygon coordinates defining the borders of the region.
vec.inout	logical. If TRUE a logical vector is included in the output indicating whether each point of the grid is inside the polygon. Defaults to FALSE.
...	currently not used (kept for back compatibility).

### Details

The function works as follows: First it creates a grid using the R function [expand.grid](#) and then it uses the `geoR`' internal function `.geoR_inout()` which wraps usage of [SpatialPoints](#) and [over](#) from the package `sp` to extract the points of the grid which are inside the polygon.

### Value

A list with components:

xypoly	an $n \times 2$ matrix with the coordinates of the points inside the polygon.
vec.inout	logical, a vector indicating whether each point of the rectangular grid is inside the polygon. Only returned if <code>vec.inout = TRUE</code> .

### Author(s)

Paulo Justiniano Ribeiro Jr. <paulojus@leg.ufpr.br>,  
Peter J. Diggle <p.diggle@lancaster.ac.uk>.

### References

See the package **geoR**.

### See Also

[expand.grid](#), [over](#), [SpatialPoints](#).

**Examples**

```
poly <- matrix(c(.2, .8, .7, .1, .2, .1, .2, .7, .7, .1), ncol=2)
plot(0:1, 0:1, type="n")
lines(poly)
poly.in <- polygrid(seq(0,1,l=11), seq(0,1,l=11), poly, vec=TRUE)
points(poly.in$xy)
```

---

 polys

---

*Connect the Values with a Polygon*


---

**Description**

Connect the values for the elements with a polygon. Every "point" has his own shape and this demonstrates the characteristic of the point.

**Usage**

```
polys(x, scale = TRUE, labels = dimnames(x)[[1]], locations = NULL,
      nrow = NULL, ncol = NULL, key.loc = NULL, key.labels = dimnames(x)[[2]],
      key.xpd = TRUE, xlim = NULL, ylim = NULL, flip.labels = NULL, factx = 1,
      facty = 1, col.stars = NA, axes = FALSE, frame.plot = axes, main = NULL,
      sub = NULL, xlab = "", ylab = "", cex = 0.8, lwd = 1.1, lty = par("lty"),
      xpd = FALSE,
      mar = pmin(par("mar"), 1.1 + c(2 * axes + (xlab != ""), 2 * axes +
        (ylab != ""), 1, 0)),
      add = FALSE, plot = TRUE, ...)
```

**Arguments**

x	a matrix or a data frame
scale	if TRUE, the data will be scaled
labels	the labels for the polygons inside the map
locations	the locations for the polygons inside the map
nrow, ncol	integers giving the number of rows and columns to use when locations=NULL. By default, 'nrow==ncol', a square layout will be used.
key.loc	the location for the legend
key.labels	the labels in the legend
key.xpd	A logical value or NA. If FALSE, all plotting is clipped to the plot region, if TRUE, all plotting is clipped to the figure region, and if NA, all plotting is clipped to the device region.
flip.labels	logical indicating if the label locations should flip up and down from diagram to diagram.
factx	additive factor for the x-coordinate
facty	magnification for the influence of the x-coordinate on the y-coordinate

main, sub, xlab, ylab, xlim, ylim, col.stars, cex, lwd, lty, xpd, mar	graphical parameters and labels for the plot
axes	if FALSE, no axes will be drawn
frame.plot	if TRUE, a box will be made around the plot
add	if TRUE, it will be added to the plot
plot	nothing is plotted
...	further graphical parameters

### Details

Each polygon represents one row of the input x. For the variables the values are computed and then those values are connected with a polygon. The location of the polygons can be defined by the user.

### Value

No return value, creates a plot.

### Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

### References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

### Examples

```
data(ohorizon)
X=ohorizon[, "XC00"]
Y=ohorizon[, "YC00"]
el=log10(ohorizon[,c("Cu", "Ni", "Na", "Sr")])
sel <- c(3,8,22, 29, 32, 35, 43, 69, 73, 93,109,129,130,134,168,181,183,205,211,
        218,237,242,276,292,297,298,345,346,352,372,373,386,408,419,427,441,446,490,
        516,535,551,556,558,564,577,584,601,612,617)
x=el[sel,]
plot(X,Y,frame.plot=FALSE,xaxt="n",yaxt="n",xlab="",ylab="",type="n",
     xlim=c(360000,max(X)))
polys(x,ncol=8,key.loc=c(15,1),factx=0.30,facty=2.0,cex=0.75,lwd=1.1)
```

---

`ppplot.das`*PP plot*

---

**Description**

This function computes a PP (Probability-Probability) plot for the given dataset.

**Usage**

```
ppplot.das(x, pdist = pnorm, xlab = NULL, ylab = "Probability", line = TRUE,  
           lwd = 2, pch = 3, cex = 0.7, cex.lab = 1, ...)
```

**Arguments**

<code>x</code>	dataset
<code>pdist</code>	the distribution function
<code>xlab, ylab, lwd, pch, cex, cex.lab</code>	graphical parameters
<code>line</code>	if a regression line should be added
<code>...</code>	further parameters for the probability function

**Details**

The empirical probability is calculated and compared with the comparison distribution.

**Value**

No return value, creates a plot.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
data(AuNEW)  
ppplot.das(AuNEW, pdist=plnorm, xlab="Probability of Au",  
           ylab="Probabilities of lognormal distribution", pch=3, cex=0.7)
```



---

qplot.das

*QP plot*


---

### Description

This function produces a QP (Quantile-Probability) plot of the data.

### Usage

```
qplot.das(x, qdist = qnorm, probs = NULL, logx = FALSE, cex.lab = 1,
xlab = NULL, ylab = "Probability [%]", line = TRUE, lwd = 2, pch = 3,
logfinetick = c(10), logfinelab = c(10), cex = 0.7, xlim = NULL,
ylim = NULL, gridy = TRUE, add.plot = FALSE, col = 1, ...)
```

### Arguments

x	data
qdist	The probability function with which the data should be compared.
probs	The selected probabilities, see details
logx	if TRUE, then log scale on x-axis is used
cex.lab	The size of the label
xlab	title for x-axis
ylab	title for y-axis
line	if TRUE the line will be drawn
lwd	the width of the line
pch, cex, col	graphical parameter
logfinetick	how fine are the tick marks on log-scale on x-axis
logfinelab	how fine are the labels on log-scale on x-axis
xlim	the range for the x-axis
ylim	the range for the y-axis
gridy	if grid along y-axis should be drawn
add.plot	if TRUE the new plot is added to an old one
...	further arguments for the probability function

### Details

First the probability of the sorted input x is computed and than the selected quantiles are calculated and after that plot is produced. If probs=NULL then the 1%, 5%, 10%, 20%,..., 90%, 95% and 99% quantile is taken.

### Value

No return value, creates a plot.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**See Also**

[plot](#), [par](#), [plot.default](#)

**Examples**

```
data(AuNEW)
qqplot.das(AuNEW,qdlist=qlnorm,xlab="Au",
ylab="Probabilities of lognormal distribution", pch=3,cex=0.7)
```

---

qqplot.das

*QQ plot*

---

**Description**

A QQ (Quantile-Quantile) plot is produced.

**Usage**

```
qqplot.das(x, distribution = "norm", ylab = deparse(substitute(x)),
  xlab = paste(distribution, "quantiles"), main = "", las = par("las"),
  datax = FALSE, envelope = 0.95, labels = FALSE, col = palette()[2],
  lwd = 2, pch = 1, line = c("quartiles", "robust", "none"), cex = 1,
  xaxt = "s", add.plot=FALSE,xlim=NULL,ylim=NULL,...)
```

**Arguments**

x	numeric vector
distribution	name of the comparison distribution
ylab	label for the y axis (empirical quantiles)
xlab	label for the x axis (comparison quantiles)
main	title for the plot
las	if 0, ticks labels are drawn parallel to the axis
datax	if TRUE, x and y axis are exchanged
envelope	confidence level for point-wise confidence envelope, or FALSE for no envelope
labels	vector of point labels for interactive point identification, or FALSE for no labels

col, lwd, pch, cex, xaxt	graphical parameter, see par
line	"quartiles" to pass a line through the quartile-pairs, or "robust" for a robust-regression line. "none" suppresses the line
add.plot	if TRUE the new plot is added to an old one
xlim	the range for the x-axis
ylim	the range for the y-axis
...	further arguments for the probability function

### Details

The probability of the input data is computed and with this result the quantiles of the comparison distribution are calculated. If line="quartiles" a line based on quartiles is plotted and if line="robust" a robust LM model is calculated.

### Value

No return value, creates a plot.

### Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

### References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

### See Also

[par](#)

### Examples

```
data(AuNEW)
qqplot.das(AuNEW,distribution="lnorm",col=1,envelope=FALSE,datax=TRUE,ylab="Au",
xlab="Quantiles of lognormal distribution", main="",line="none",pch=3,cex=0.7)
```

---

res.eyefit.As\_C      *Result of the function eyefit for variogram estimation.*

---

### Description

This result could also be directly computed using the function eyefit.

### Usage

```
data(res.eyefit.As_C)
```

### Format

The format is: List of 1 \$ :List of 7 ..\$ cov.model: chr "spherical" ..\$ cov.pars : num [1:2] 0.8 160.3 ..\$ nugget : num 0.49 ..\$ kappa : num 0.5 ..\$ lambda : num 0 ..\$ trend : chr "cte" ..\$ max.dist : num 288 ..- attr(\*, "class")= chr "variomodel" - attr(\*, "class")= chr "eyefit"

### Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

### References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

### Examples

```
data(res.eyefit.As_C)
str(res.eyefit.As_C)
```

---

res.eyefit.As\_C\_m      *Result of the function eyefit for variogram estimation.*

---

### Description

This result could also be directly computed using the function eyefit.

### Usage

```
data(res.eyefit.As_C_m)
```

### Format

The format is: List of 1 \$ :List of 7 ..\$ cov.model: chr "spherical" ..\$ cov.pars : num [1:2] 0.8 160255.8 ..\$ nugget : num 0.49 ..\$ kappa : num 0.5 ..\$ lambda : num 0 ..\$ trend : chr "cte" ..\$ max.dist : num 288460 ..- attr(\*, "class")= chr "variomodel" - attr(\*, "class")= chr "eyefit"

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
data(res.eyefit.As_C_m)
str(res.eyefit.As_C_m)
```

---

res.eyefit.AuNEW      *Result of the function eyefit for variogram estimation.*

---

**Description**

This result could also be directly computed using the function eyefit.

**Usage**

```
data(res.eyefit.AuNEW)
```

**Format**

The format is: List of 1 \$ :List of 7 ..\$ cov.model: chr "exponential" ..\$ cov.pars : num [1:2] 0.31 53418.46 ..\$ nugget : num 0.44 ..\$ kappa : num 0.5 ..\$ lambda : num 0 ..\$ trend : chr "cte" ..\$ max.dist : num 192306 ..- attr(\*, "class")= chr "variomodel" - attr(\*, "class")= chr "eyefit"

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
data(res.eyefit.AuNEW)
str(res.eyefit.AuNEW)
```

---

res.eyefit.Ca\_C      *Result of the function eyefit for variogram estimation.*

---

### Description

This result could also be directly computed using the function eyefit.

### Usage

```
data(res.eyefit.Ca_C)
```

### Format

The format is: List of 1 \$ :List of 7 ..\$ cov.model: chr "spherical" ..\$ cov.pars : num [1:2] 3.80e-01 1.92e+05 ..\$ nugget : num 0.21 ..\$ kappa : num 0.5 ..\$ lambda : num 0 ..\$ trend : chr "cte" ..\$ max.dist : num 192306 ..- attr(\*, "class")= chr "variomodel" - attr(\*, "class")= chr "eyefit"

### Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

### References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

### Examples

```
data(res.eyefit.Ca_C)
str(res.eyefit.Ca_C)
```

---

res.eyefit.Ca\_0      *Result of the function eyefit for variogram estimation.*

---

### Description

This result could also be directly computed using the function eyefit.

### Usage

```
data(res.eyefit.Ca_0)
```

### Format

The format is: List of 1 \$ :List of 7 ..\$ cov.model: chr "spherical" ..\$ cov.pars : num [1:2] 0.01 5341.85 ..\$ nugget : num 0.12 ..\$ kappa : num 0.5 ..\$ lambda : num 0 ..\$ trend : chr "cte" ..\$ max.dist : num 192306 ..- attr(\*, "class")= chr "variomodel" - attr(\*, "class")= chr "eyefit"

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
data(res.eyefit.Ca_0)
str(res.eyefit.Ca_0)
```

---

res.eyefit.Hg\_0      *Result of the function eyefit for variogram estimation.*

---

**Description**

This result could also be directly computed using the function eyefit.

**Usage**

```
data(res.eyefit.Hg_0)
```

**Format**

The format is: List of 1 \$ :List of 7 ..\$ cov.model: chr "exponential" ..\$ cov.pars : num [1:2] 1.50e-02 3.21e+04 ..\$ nugget : num 0.04 ..\$ kappa : num 0.5 ..\$ lambda : num 0 ..\$ trend : chr "cte" ..\$ max.dist : num 288460 ..- attr(\*, "class")= chr "variomodel" - attr(\*, "class")= chr "eyefit"

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
data(res.eyefit.Hg_0)
str(res.eyefit.Hg_0)
```

---

```
res.eyefit.Pb_01      Result of the function eyefit for variogram estimation.
```

---

**Description**

This result could also be directly computed using the function `eyefit`.

**Usage**

```
data(res.eyefit.Pb_01)
```

**Format**

The format is: List of 1 \$ :List of 7 ..\$ cov.model: chr "spherical" ..\$ cov.pars : num [1:2] 1.90e-01 5.13e+05 ..\$ nugget : num 0.11 ..\$ kappa : num 0.5 ..\$ lambda : num 0 ..\$ trend : chr "cte" ..\$ max.dist : num 288460 ..- attr(\*, "class")= chr "variomodel" - attr(\*, "class")= chr "eyefit"

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
data(res.eyefit.Pb_01)
str(res.eyefit.Pb_01)
```

---

```
res.eyefit.Pb_02      Result of the function eyefit for variogram estimation.
```

---

**Description**

This result could also be directly computed using the function `eyefit`.

**Usage**

```
data(res.eyefit.Pb_02)
```

**Format**

The format is: List of 1 \$ :List of 7 ..\$ cov.model: chr "spherical" ..\$ cov.pars : num [1:2] 0.03 48076.64 ..\$ nugget : num 0.11 ..\$ kappa : num 0.5 ..\$ lambda : num 0 ..\$ trend : chr "cte" ..\$ max.dist : num 288460 ..- attr(\*, "class")= chr "variomodel" - attr(\*, "class")= chr "eyefit"



**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
data(res.eyefit.Pb_02)
str(res.eyefit.Pb_02)
```

---

rg.boxplot	<i>Plot a Boxplot</i>
------------	-----------------------

---

**Description**

Plot a single horizontal boxplot, the default is a Tukey boxplot.

**Usage**

```
rg.boxplot(xx, xlab = deparse(substitute(xx)), log = FALSE, ifbw = FALSE,
wend = 0.05, xlim = NULL, main = " ", colr = 5, ...)
```

**Arguments**

xx	data
xlab	label for the x-axis
log	if TRUE, a log-scaled plot and a logtransformation of the data
ifbw	if TRUE, a IDEAS style box-and-whisker plot is produced
wend	defines the end of the whisker, default is 5% and 95% quantile
xlim	setting xlim results in outliers not being plotted as the x-axis is shortened.
main	main title of the plot
colr	the box is infilled with a yellow ochre; if no colour is required set colr=0
...	further graphical parameters for the plot

**Details**

As the x-axis is shortend by setting xlim, however, the statistics used to define the boxplot, or box-and-whisker plot, are still based on the total data set. To plot a truncated data set create a subset first, or use the `x[x<some.value]` construct in the call.

**Value**

No return value, creates a plot.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
data(chorizon)
Ba=chorizon[, "Ba"]
rg.boxplot(Ba, ifbw=TRUE, colr=0, xlab="Ba [mg/kg]", cex.lab=1.2)
```

---

 rg.mva

*Non-robust Multivariate Data Analysis*


---

**Description**

Procedure to undertake non-robust multivariate data analysis. The saved list may be passed to other rotation and display functions

**Usage**

```
rg.mva(x, main = deparse(substitute(x)))
```

**Arguments**

x	data
main	used for the list

**Details**

Procedure to undertake non-robust multivariate data analyses; the object generated is identical to that of `rg.robmva` so that the savedlist may be passed to other rotation and display functions. Thus weights are set to 1, and other variables are set to appropriate defaults. The estimation of Mahalanobis distances is only undertaken if `x` is nonsingular, i.e. the lowest eigenvalue is  $> 10e-4$ .

**Value**

n	number of rows
p	number of columns
wts	the weights for the covariance matrix
mean	the mean of the data
cov	the covariance
sd	the standard deviation

r	correlation matrix
eigenvalues	eigenvalues of the SVD
econtrib	proportion of eigenvalues in %
eigenvectors	eigenvectors of the SVD
rload	loadings matrix
rcr	standardised loadings matrix
vcontrib	scores variance
pvcontrib	proportion of scores variance in %
cpvcontrib	cummulative proportion of scores variance
md	Mahalanbois distance
ppm	probability for outliegness using F-distribution
epm	probability for outliegness using Chisquared-distribution

### Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

### References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

### Examples

```
#input data
data(ohorizon)
vegzn=ohorizon[, "VEG_ZONE"]
veg=rep(NA,nrow(ohorizon))
veg[vegzn=="BOREAL_FOREST"] <- 1
veg[vegzn=="FOREST_TUNDRA"] <- 2
veg[vegzn=="SHRUB_TUNDRA"] <- 3
veg[vegzn=="DWARF_SHRUB_TUNDRA"] <- 3
veg[vegzn=="TUNDRA"] <- 3
el=c("Ag", "Al", "As", "B", "Ba", "Bi", "Ca", "Cd", "Co", "Cu", "Fe", "K", "Mg", "Mn",
      "Na", "Ni", "P", "Pb", "Rb", "S", "Sb", "Sr", "Th", "Tl", "V", "Y", "Zn")
x <- log10(ohorizon[!is.na(veg),el])
v <- veg[!is.na(veg)]

rg.mva(as.matrix(x[v==1,]))
```

---

 rg.mvalloc

*Robust Multivariate Allocation Procedure*


---

**Description**

Function to allocate an individual to one of several populations.

**Usage**

```
rg.mvalloc(pcrit = 0.05, x, ...)
```

**Arguments**

pcrit	When the probability of group membership is less than pcrit it is allocated to group 0.
x	contains the individuals to be allocated
...	arguments for creating a list of groups

**Details**

m objects are the reference populations generated by md.gait, rg.robmva or rg.mva to estimate Mahalanobis distances and predicted probabilities of group membership for individuals in matrix x. Note that the log |determinant| of the appropriate covariance matrix is added to the Mahalanobis distance on the assumption that the covariance matrices are inhomogeneous. If the data require transformation this must be undertaken before calling this function. This implies that a similar transformation must have been used for all the reference data subsets.

**Value**

groups	the groups
m	number of groups
n	number of individuals to be allocated
p	number of columns
pgm	number of individuals to be allocated multiplied with the groups
pcrit	critical probability
xalloc	number of individuals as integer

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
#input data
data(ohorizon)
vegzn=ohorizon[, "VEG_ZONE"]
veg=rep(NA,nrow(ohorizon))
veg[vegzn=="BOREAL_FOREST"] <- 1
veg[vegzn=="FOREST_TUNDRA"] <- 2
veg[vegzn=="SHRUB_TUNDRA"] <- 3
veg[vegzn=="DWARF_SHRUB_TUNDRA"] <- 3
veg[vegzn=="TUNDRA"] <- 3
el=c("Ag", "Al", "As", "B", "Ba", "Bi", "Ca", "Cd", "Co", "Cu", "Fe", "K", "Mg", "Mn",
      "Na", "Ni", "P", "Pb", "Rb", "S", "Sb", "Sr", "Th", "Tl", "V", "Y", "Zn")
x <- log10(ohorizon[!is.na(veg),el])
v <- veg[!is.na(veg)]

res.zone1=rg.mva(as.matrix(x[v==1,]))
res.zone2=rg.mva(as.matrix(x[v==2,]))
res.zone3=rg.mva(as.matrix(x[v==3,]))
res=rg.mvalloc(pcrit=0.01,x,res.zone1,res.zone2,res.zone3)
```

---

 rg.remove.na

*Remove NA*


---

**Description**

Function to remove NAs from a vector and inform the user of how many.

**Usage**

```
rg.remove.na(xx)
```

**Arguments**

xx                      vector

**Details**

The function counts the NAs in a vector and returns the number of NAs and the "new" vector.

**Value**

x                        vector without the NAs  
 nna                      number of NAs removed

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

## References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

## Examples

```
x<-rep(NA,10)
x[c(1,3,5,7,9)]<-10
rg.remove.na(x)
```

---

 rg.robmva

*Robust Multivariate Analysis*


---

## Description

Procedure for multivariate analysis using the minimum volume ellipsoid (MVE), minimum covariance determinant (MCD) or a supplied set of 0-1 weights.

## Usage

```
rg.robmva(x, proc = "mcd", wts = NULL, main = deparse(substitute(x)))
```

## Arguments

x	data
proc	procedure for the estimation (MVE or MCD)
wts	if proc=NULL, the supplied weights for the calculation
main	input for the list

## Details

cov.mcd is limited to a maximum of 50 variables. Both of these procedures lead to a vector of 0-1 weights and mcd is the default. A set of weights can be generated by using Graphical Adaptive Interactive Trimming (GAIT) procedure available through rg.md.gait(). Using 0-1 weights the parameters of the background distribution are estimated by cov.wt(). A robust estimation of the Mahalanobis distances is made for the total data set but is only undertaken if x is non-singular (lowest eigenvalue is >10e-4).

## Value

n	number of rows
p	number of columns
wts	the weights for the covariance matrix
mean	the mean of the data
cov	the covariance

sd	the standard deviation
r	correlation matrix
eigenvalues	eigenvalues of the SVD
econtrib	proportion of eigenvalues in %
eigenvectors	eigenvectors of the SVD
rload	loadings matrix
rcr	standardised loadings matrix
vcontrib	scores variance
pvcontrib	proportion of scores variance in %
cpvcontrib	cummulative proportion of scores variance
md	Mahalanbois distance
ppm	probability for outliegness using F-distribution
epm	probability for outliegness using Chisquared-distribution

### Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

### References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

### Examples

```
#input data
data(ohorizon)
vegzn=ohorizon[, "VEG_ZONE"]
veg=rep(NA,nrow(ohorizon))
veg[vegzn=="BOREAL_FOREST"] <- 1
veg[vegzn=="FOREST_TUNDRA"] <- 2
veg[vegzn=="SHRUB_TUNDRA"] <- 3
veg[vegzn=="DWARF_SHRUB_TUNDRA"] <- 3
veg[vegzn=="TUNDRA"] <- 3
e1=c("Ag", "Al", "As", "B", "Ba", "Bi", "Ca", "Cd", "Co", "Cu", "Fe", "K", "Mg", "Mn",
     "Na", "Ni", "P", "Pb", "Rb", "S", "Sb", "Sr", "Th", "Tl", "V", "Y", "Zn")
x <- log10(ohorizon[!is.na(veg),e1])
v <- veg[!is.na(veg)]
subvar=c("Ag", "B", "Bi", "Mg", "Mn", "Na", "Pb", "Rb", "S", "Sb", "Tl")
set.seed(100)

rg.robmva(as.matrix(x[v==1,subvar]))
```

---

`rg.wtdsums`*Calculate Weighted Sums for a Matrix*

---

**Description**

This function computes a weighted sum for a matrix based on computed quantiles and user defined relative importance.

**Usage**

```
rg.wtdsums(x, ri, xcentr = NULL, xdisp = NULL)
```

**Arguments**

<code>x</code>	matrix
<code>ri</code>	vector for the relative importance, $\text{length}(ri)=\text{length}(x[1,])$
<code>xcentr</code>	the provided center
<code>xdisp</code>	the provided variance

**Details**

It is not necessary to provide the center and the variance. If those values are not supplied the center is the 50% quantile and the variance is calculated from the 25% and 75% quantile.

**Value**

<code>input</code>	input parameter
<code>centr</code>	the center
<code>disp</code>	the variance
<code>ri</code>	relative importance
<code>w</code>	weights
<code>a</code>	normalized weights
<code>ws</code>	normalized weights times standardized x

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.



**Examples**

```
data(chorizon)
var=c("Si_XRF","Al_XRF","K_XRF","LOI","P","Mn")
ri=c(-2.0,1.5,2.0,2.0,3.0,2.0)
x=chorizon[,var]
rg.wtdsums(x,ri)
```

---

RobCor.plot

*Compares the Robust Estimation with the Classical*


---

**Description**

This function compares a robust covariance (correlation) estimation (MCD is used) with the classical approach. A plot with the two ellipses will be produced and the correlation coefficients are quoted.

**Usage**

```
RobCor.plot(x, y, quan = 1/2, alpha = 0.025, colC = 1, colR = 1, ltyC = 2,
ltyR = 1, ...)
```

**Arguments**

x, y	two data vectors where the correlation should be computed
quan	fraction of tolerated outliers (at most 0.5)
alpha	quantile of chisquare distribution for outlier cutoff
colC, colR	colour for both ellipses
ltyC, ltyR	line type for both ellipses
...	other graphical parameters

**Details**

The covariance matrix is estimated in a robust (MCD) and non robust way and then both ellipses are plotted. The radi is calculated from the singular value decomposition and a breakpoint (specified quantile) for outlier cutoff.

**Value**

cor.cla	correlation of the classical estimation
cor.rob	correlation of the robust estimation

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

## References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

## Examples

```
data(chorizon)
Be=chorizon[,"Be"]
Sr=chorizon[,"Sr"]
RobCor.plot(log10(Be),log10(Sr),xlab="Be in C-horizon [mg/kg]",
ylab="Sr in C-horizon [mg/kg]",cex.lab=1.2, pch=3, cex=0.7,
xaxt="n", yaxt="n",colC=1,colR=1,ltyC=2,ltyR=1)
```

---

roundpretty

*Roundpretty*

---

## Description

Round a value in a pretty way.

## Usage

```
roundpretty(kvec, maxdig)
```

## Arguments

kvec	the variable to be rounded
maxdig	maximum number of digits after the coma

## Value

result	rounded value
--------	---------------

## Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

## References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

## See Also

[roundpretty.sub](#)

**Examples**

```
roundpretty(0.873463029,5)
roundpretty(0.073463029,5)
roundpretty(0.003463029,5)
roundpretty(0.000463029,5)
```

---

roundpretty.sub	<i>Subfunction for Roundpretty</i>
-----------------	------------------------------------

---

**Description**

This function rounds the number in pretty way.

**Usage**

```
roundpretty.sub(k, maxdig)
```

**Arguments**

k	number to be rounded pretty
maxdig	maximum number of digits after the coma

**Details**

When maxdig is larger than 8 and the number is smaller than 0.00001, the number is rounded to 8 numbers after the coma. When the number is smaller than 0.0001 the maximum numbers after the coma is 7, and so on.

**Value**

kr	rounded value
----	---------------

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**See Also**

[roundpretty](#)

---

`scalebar`*Scalebar*

---

**Description**

This function plots the unit at a specified location.

**Usage**

```
scalebar(Xlowerleft, Ylowerleft, Xupperright, Yupperright, shifttext, shiftkm,
sizetext)
```

**Arguments**

<code>Xlowerleft, Ylowerleft</code>	x and y coordinate of the lower left corner
<code>Xupperright, Yupperright</code>	x and y coordinate of the upper corner
<code>shifttext</code>	on which margin line, starting at 0 counting outwards
<code>shiftkm</code>	how far from the last point the label should be written
<code>sizetext</code>	expansion factor for the text

**Value**

No return value, creates a plot.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
plot.new()
scalebar(0,0.25,1,0.5,shifttext=-0.05,shiftkm=4e4,sizetext=0.8)
```

---

scatter3dPETER

*3D plot of a Regression Model*


---

### Description

This function makes a 3D plot of the data and the regression function. The user has the choice between different methods to calculate the coefficients for the regression model.

### Usage

```
scatter3dPETER(x, y, z, xlab = deparse(substitute(x)),
ylab = deparse(substitute(y)), zlab = deparse(substitute(z)),
revolutions = 0, bg.col = c("white", "black"),
axis.col = if (bg.col == "white") "black" else "white",
surface.col = c("blue", "green", "orange", "magenta", "cyan", "red",
"yellow", "gray"), neg.res.col = "red",
pos.res.col = "green", point.col = "yellow", text.col = axis.col,
grid.col = if (bg.col == "white") "black" else "gray",
fogtype = c("exp2", "linear", "exp", "none"),
residuals = (length(fit) == 1), surface = TRUE, grid = TRUE,
grid.lines = 26, df.smooth = NULL, df.additive = NULL, sphere.size = 1,
threshold = 0.01, speed = 1, fov = 60, fit = "linear", groups = NULL,
parallel = TRUE, model.summary = FALSE)
```

### Arguments

x, y, z	the coordinates for the points
xlab, ylab, zlab	the labels for the axis
revolutions	if the plot should be viewed from different angles
bg.col, axis.col, surface.col, point.col, text.col, grid.col	define the colour for the background, axis,...
pos.res.col, neg.res.col	colour for positive and negativ residuals
fogtype	describes the fogtype, see rgl.bg
residuals	if the residuals should be plotted
surface	if the regression function should be plotted or just the points
grid	if TRUE, the grid is plotted
grid.lines	number of lines in the grid
df.smooth	if fit=smooth, the number of degrees of freedom
df.additive	if fit=additive, the number of degrees of freedom
sphere.size	a value for calibrating the size of the sphere
threshold	the minimum size of the sphere, if the size is smaller than the threshold a point is plotted

speed	if revolutions>0, how fast you make a 360 degree turn
fov	field-of-view angle, see rgl.viewpoint
fit	which method should be used for the model; "linear", "quadratic", "smooth" or "additive"
groups	define groups for the points
parallel	if groups is not NULL, a parallel shift in the model is made
model.summary	if the summary should be returned

### Details

The user can choose between a linear, quadratic, smoothed or additive model to calculate the coefficients.

### Value

No return value, creates a plot.

### Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

### References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

### Examples

```
#required library
#require(IPSUR)
data(chorizon)
lit=1
# This example needs additional libraries:
#scatter3dPETER(x=log10(chorizon[chorizon$LIT0==lit,"Cr"]),
#              z=log10(chorizon[chorizon$LIT0==lit,"Cr_INAA"]),
#              y=log10(chorizon[chorizon$LIT0==lit,"Co"]),
#              xlab="",ylab="",zlab="",
#              neg.res.col=gray(0.6), pos.res.col=gray(0.1), point.col=1, fov=30,
#              surface.col="black",grid.col="gray",sphere.size=0.8)
```

**Description**

Plots smoothing maps and legend based on continuous or percentile scale.

**Usage**

```
SmoothLegend(X, Y, z, resol = 200, type = "percentile", whichcol = "gray",
  qtiles = c(0, 0.05, 0.25, 0.5, 0.75, 0.9, 0.95, 1), borders=NULL, leg.xpos.min = 780000,
  leg.xpos.max = 8e+05, leg.ypos.min = 7760000, leg.ypos.max = 7870000,
  leg.title = "mg/kg", leg.title.cex = 0.7, leg.numb.cex = 0.7, leg.round = 2,
  leg.wid = 4, leg.numb.xshift = 70000, leg.perc.xshift = 40000,
  leg.perc.yshift = 20000, tit.xshift = 35000)
```

**Arguments**

X	X-coordinates
Y	Y-coordinates
z	values on the coordinates
resol	resolution of smoothing
type	"percentile" for percentile legend; "contin" for continuous grey-scale or colour map
whichcol	type of color scheme to use: "grey", "rainbow", "rainbow.trunc", "rainbow.inv", "terrain" or "topo"
qtiles	considered quantiles if type="percentile" is used
borders	either NULL or character string with the name of the list with list elements x and y for x- and y-coordinates of map borders
leg.xpos.min	minimum value of x-position of the legend
leg.xpos.max	maximum value of x-position of the legend
leg.ypos.min	minimum value of y-position of the legend
leg.ypos.max	maximum value of y-position of the legend
leg.title	title for legend
leg.title.cex	cex for legend title
leg.numb.cex	cex for legend numbers
leg.round	round legend to specified digits "pretty"
leg.wid	width (space in numbers) for legend
leg.numb.xshift	x-shift of numbers in legend relative to leg.xpos.max
leg.perc.xshift	x-shift of "Percentile" in legend relative to leg.xpos.min

```
leg.perc.yshift      y-shift of "Percentile" in legend relative to leg.ypos.max
tit.xshift          x-shift of title in legend relative to leg.xpos.max
```

### Details

First a interpolation is applied using different versions of algorithms from Akima and then all points a distinguished into inside an outside the polygonal region. Now the empirical quantiles for points inside the polygon are computed and then the values are plotted in different scales of the choosen colour. ATTENTION: here borders were defined for the smoothing region

### Value

No return value, creates a plot.

### Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

### References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

### Examples

```
data(chorizon)
X=chorizon[,"XC00"]
Y=chorizon[,"YC00"]
el=log10(chorizon[,"As"])

# generate plot
plot(X,Y,frame.plot=FALSE,xaxt="n",yaxt="n",xlab="",ylab="",type="n")

data(bordersKola) # list with list elements x and y for x- and y-coordinates of map borders
SmoothLegend(X,Y,el,resol=200,type="contin",whichcol="gray",
  qtiles=c(0,0.05,0.25,0.50,0.75,0.90,0.95,1), borders="bordersKola",
  leg.xpos.min=7.8e5,leg.xpos.max=8.0e5,leg.ypos.min=77.6e5,leg.ypos.max=78.7e5,
  leg.title="mg/kg", leg.title.cex=0.7, leg.numb.cex=0.7, leg.round=2,leg.wid=4,
  leg.numb.xshift=0.7e5,leg.perc.xshift=0.4e5,leg.perc.yshift=0.2e5,tit.xshift=0.35e5)

# plot background
data(kola.background)
plotbg(map.col=c("gray","gray","gray","gray"),map.lwd=c(1,1,1,1),add.plot=TRUE)
```



---

suns *Plot Suns*

---

### Description

This function makes a graphical diagram of multivariate data. Every element represents one line in the sun and the length of the line indicates the concentration of the element.

### Usage

```
suns(x, full = TRUE, scale = TRUE, radius = TRUE, labels = dimnames(x)[[1]],
locations = NULL, nrow = NULL, ncol = NULL, len = 1, key.loc = NULL,
key.labels = dimnames(x)[[2]], key.xpd = TRUE, xlim = NULL, ylim = NULL,
flip.labels = NULL, col.stars = NA, axes = FALSE, frame.plot = axes, main = NULL,
sub = NULL, xlab = "", ylab = "", cex = 0.8, lwd = 0.25, lty = par("lty"),
xpd = FALSE,
mar = pmin(par("mar"), 1.1 + c(2 * axes + (xlab != ""), 2 * axes + (ylab != ""), 1, 0)),
add = FALSE, plot = TRUE, ...)
```

### Arguments

x	a matrix or a data frame
full	if TRUE, a whole circle will be made
scale	if TRUE, the data will be scaled
radius	should be TRUE, otherwise the lines in the sun will not be plotted
labels	the labels for the suns inside the map
locations	the locations for the suns inside the map
nrow, ncol	integers giving the number of rows and columns to use when locations=NULL
len	scaling factor for the length of the lines (according to the size of the map)
key.loc	the location for the legend
key.labels	the labels in the legend
key.xpd	A logical value or NA. If FALSE, all plotting is clipped to the plot region, if TRUE, all plotting is clipped to the figure region, and if NA, all plotting is clipped to the device region.
flip.labels	logical indication if the label locations should flip up and down from diagram to diagram.
axes	if FALSE, no axes will be drawn
frame.plot	if TRUE, a box will be made around the plot
main, sub, xlab, xlim, ylim, col.stars, ylab, cex, lwd, lty, xpd, mar	graphical parameters and labels for the plot
add	if TRUE, it will be added to the plot
plot	nothing is plotted
...	graphical parameters for plotting the box

**Details**

Each sun represents one row of the input x. Each line of the sun represents one chosen element. The distance from the center of the sun to the point shows the size of the value of the (scaled) column.

**Value**

No return value, creates a plot.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
data(ohorizon)
X=ohorizon[,"XC00"]
Y=ohorizon[,"YC00"]
el=log10(ohorizon[,c("Co", "Cu", "Ni", "Rb", "Bi", "Na", "Sr")])

sel <- c(3,8,22, 29, 32, 35, 43, 69, 73 ,93,109,129,130,134,168,181,183,205,211,
        218,237,242,276,292,297,298,345,346,352,372,373,386,408,419,427,441,446,490,
        516,535,551,556,558,564,577,584,601,612,617)
x=el[sel,]
suns(x,ncol=8,key.loc=c(15,0.5),lwd=1.3)
```

---

SymbLegend

*Plot Legend*

---

**Description**

Plots symbols and Legend on a map. There are two different methods (percentile symbols or boxplot symbols) to display the legend.

**Usage**

```
SymbLegend(X, Y, z, type = "percentile", qtiles = c(0, 0.05, 0.25, 0.75, 0.95, 1),
q = NULL, symbtype = "EDA", symbmagn = 0.8, leg.position = "topright",
leg.title = "", leg.title.cex = 0.8, leg.round = 2, leg.wid = 4, leg.just = "right",
cex.scale = 0.8, xf = 9000, logscale = TRUE, accentuate = FALSE)
```

**Arguments**

X	X-coordinates
Y	Y-coordinates
z	values on the coordinates
type	"percentile" for percentile legend, "boxplot" for boxplot legend
quintiles	considered quantiles if type="percentile" is used
q	if not NULL, provide manually data points where to break
symbtype	type of symbols to be used; "EDA", "EDAacc", "EDAacc2", "EDAext", "GSC" or "arbit"
symbmagn	magnification factor for symbols
leg.position	position of the legend, either character like "topright" or coordinates
leg.title	title for legend
leg.title.cex	cex for legend
leg.round	round legend to specified digits "pretty"
leg.wid	width (space in numbers) for legend
leg.just	how to justify the legend
cex.scale	cex for text "log-scale" and for boxplot legend - only for type="boxplot"
xf	x-distance from boxplot to number for legend
logscale	if TRUE a log scale is used (for boxplot scale) and the log-boxplot is computed
accentuate	if TRUE, accentuated symbols are used (here only EDA accentuated!)

**Details**

It is possible to choose between different methods for calculating the range of the values for the different symbols.

If type="percentile" the pre-determined quantiles of the data are computed and are used to plot the symbols. If type="boxplot" a boxplot is computed and the values were taken to group the values for the plot and the legend. In the case that a log scale is used the function boxplotlog is used instead of boxplot.

**Value**

No return value, creates a plot.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```

data(chorizon)
data(kola.background)
el=chorizon[, "As"]
X=chorizon[, "XCO0"]
Y=chorizon[, "YCO0"]

plot(X,Y,frame.plot=FALSE,xaxt="n",yaxt="n",xlab="",ylab="",type="n")
plotbg(map.col=c("gray","gray","gray","gray"),add.plot=TRUE)

Symblegend(X,Y,el,type="percentile",qtiles<-c(0,0.05,0.25,0.75,0.95,1),symbtype="EDA",
symbmagn=0.8,leg.position="topright",leg.title="As [mg/kg]",leg.title.cex=0.8,leg.round=2,
leg.wid=4,leg.just="right")

```

---

ternary

*Ternary plot*


---

**Description**

This plot shows the relative proportions of three variables in one diagram. It is important that the proportion sum up to 100% and if the values of the variables are very different it is important to scale them to the same data range.

**Usage**

```
ternary(x, nam = NULL, grid = FALSE, ...)
```

**Arguments**

x	matrix with 3 columns
nam	names of the variables
grid	if TRUE the grid should be plotted
...	further graphical parameters, see par

**Details**

The relative proportion of each variable is computed and those points are plotted into the graphic.

**Value**

No return value, creates a plot.

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**References**

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
data(moss)
x=moss[,c("Ni", "Cu", "Pb")]
ternary(x, grid=TRUE, pch=3, cex=0.7, col=1)
```

---

timetrend

*Data for computing time trends*

---

**Description**

These are time trends from the Kola Project data.

**Usage**

```
data(timetrend)
```

**Format**

A data frame with 96 observations on the following 47 variables.

DD a numeric vector  
MM a numeric vector  
YY a numeric vector  
Year a numeric vector  
Catch a numeric vector  
X.ID a numeric vector  
Ag a numeric vector  
Al a numeric vector  
As a numeric vector  
B a numeric vector  
Ba a numeric vector  
Be a numeric vector  
Bi a numeric vector  
Cd a numeric vector  
Co a numeric vector  
Cr a numeric vector  
Cu a numeric vector

Fe a numeric vector  
K a numeric vector  
Li a numeric vector  
Mn a numeric vector  
Mo a numeric vector  
Ni a numeric vector  
Pb a numeric vector  
Rb a numeric vector  
Sb a numeric vector  
Se a numeric vector  
Sr a numeric vector  
Th a numeric vector  
Tl a numeric vector  
U a numeric vector  
V a numeric vector  
Zn a numeric vector  
Ca a numeric vector  
Mg a numeric vector  
Na a numeric vector  
P a numeric vector  
S a numeric vector  
Si a numeric vector  
PO4 a numeric vector  
Br a numeric vector  
Cl a numeric vector  
F a numeric vector  
NO3 a numeric vector  
SO4 a numeric vector  
pH a numeric vector  
EC a numeric vector

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**Source**

Kola Project (1993-1998)

## References

Reimann C, Jørgensen M, Chekushin V, Bogatyrev I, Boyd R, Caritat P de, Dutter R, Finne TE, Halleraker JH, Jørgensen J, Kashulina G, Lehto O, Niskavaara H, Pavlov V, Røisønen ML, Strand T, Volden T. Environmental Geochemical Atlas of the Central Barents Region. NGU-GTK-CKE Special Publication, Geological Survey of Norway, Trondheim, Norway, 1998.

## Examples

```
data(timetrend)
str(timetrend)
```

---

topsoil	<i>topsoil layer of the Kola Data</i>
---------	---------------------------------------

---

## Description

The Kola Data were collected in the Kola Project (1993-1998, Geological Surveys of Finland (GTK) and Norway (NGU) and Central Kola Expedition (CKE), Russia). More than 600 samples in five different layers were analysed, this dataset contains the C-horizon.

## Usage

```
data(topsoil)
```

## Format

A data frame with 607 observations on the following 45 variables.

ID a numeric vector  
 XCOO a numeric vector  
 YCOO a numeric vector  
 ELEV a numeric vector  
 COUN a factor with levels FIN NOR RUS  
 ASP a factor with levels E FLAT N NE NW S SE SW W  
 TOPC a numeric vector  
 LITO a numeric vector  
 Ac\_228 a numeric vector  
 As a numeric vector  
 Au a numeric vector  
 Ba a numeric vector  
 Bi\_214 a numeric vector  
 Br a numeric vector  
 Ca a numeric vector

Ce a numeric vector  
Co a numeric vector  
Cr a numeric vector  
Cs a numeric vector  
Cs\_137 a numeric vector  
EC a numeric vector  
Eu a numeric vector  
Fe a numeric vector  
Hf a numeric vector  
Hg a numeric vector  
K\_40 a numeric vector  
La a numeric vector  
LOI a numeric vector  
Lu a numeric vector  
Mo a numeric vector  
Na a numeric vector  
Nd a numeric vector  
Ni a numeric vector  
pH a numeric vector  
Rb a numeric vector  
Sb a numeric vector  
Sc a numeric vector  
Sm a numeric vector  
Sr a numeric vector  
Tb a numeric vector  
Th a numeric vector  
U a numeric vector  
W a numeric vector  
Yb a numeric vector  
Zn a numeric vector

**Author(s)**

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

**Source**

Kola Project (1993-1998)



## References

Reimann C, Jørgensen M, Chekushin V, Bogatyrev I, Boyd R, Caritat P de, Dutter R, Finne TE, Halleraker JH, Jørgensen M, Kashulina G, Lehto O, Niskavaara H, Pavlov V, Røislien ML, Strand T, Volden T. Environmental Geochemical Atlas of the Central Barents Region. NGU-GTK-CKE Special Publication, Geological Survey of Norway, Trondheim, Norway, 1998.

## Examples

```
data(topsoil)
str(topsoil)
```

---

varcomp	<i>Variance Components</i>
---------	----------------------------

---

## Description

This function estimates the variance components for ANOVA.

## Usage

```
varcomp(a1, a2, f1, f2)
```

## Arguments

a1, a2	analytical duplicates
f1, f2	field duplicates

## Value

pct.regional	percentage of regional variability
pct.site	percentage at site variability
pct.analytical	percentage of analytical variability
pval	p-value

## Author(s)

Peter Filzmoser <<P.Filzmoser@tuwien.ac.at>> <http://cstat.tuwien.ac.at/filz/>

## References

C. Reimann, P. Filzmoser, R.G. Garrett, and R. Dutter: Statistical Data Analysis Explained. Applied Environmental Statistics with R. John Wiley and Sons, Chichester, 2008.

**Examples**

```
# field duplicates:
data(CHorFieldDUP)
xfield1=CHorFieldDUP[,5:98]
xfield2=CHorFieldDUP[,99:192]

# analytical duplicates:
data(CHorANADUP)
xanal1=CHorANADUP[,3:96]
xanal2=CHorANADUP[,97:190]

varcomp(xanal1[,1],xanal2[,1],xfield1[,1],xfield2[,1])
```

# Index

## \* **aplot**

- boxplotlegend, 11
- do.ellipses, 44
- KrigeLegend, 50
- Northarrow, 57
- plotuniout, 67
- polys, 70
- scalebar, 92
- SymbLegend, 98

## \* **datasets**

- AuNEW, 4
- AuOLD, 5
- bhorizon, 5
- bordersKola, 8
- CHorANADUP, 17
- CHorFieldDUP, 23
- chorizon, 30
- CHorSTANDARD, 34
- kola.background, 49
- monch, 53
- moss, 54
- nizap, 57
- ohorizon, 58
- plotbg, 63
- res.eyefit.As\_C, 76
- res.eyefit.As\_C\_m, 76
- res.eyefit.AuNEW, 77
- res.eyefit.Ca\_C, 78
- res.eyefit.Ca\_0, 78
- res.eyefit.Hg\_0, 79
- res.eyefit.Pb\_01, 80
- res.eyefit.Pb\_02, 80
- timetrend, 101
- topsoil, 103

## \* **dplot**

- arw, 3
- concareas, 37
- concareasExampleKola, 39
- loadplot, 52

- plotelement, 64
- plotellipse, 65
- plotmvoutlier, 66
- ppplot.das, 72
- qqplot.das, 73
- qqplot.das, 74
- rg.boxplot, 81
- scatter3dPETER, 93
- SmoothLegend, 95
- suns, 97

## \* **methods**

- rg.remove.na, 85
- roundpretty, 90
- roundpretty.sub, 91

## \* **models**

- scatter3dPETER, 93

## \* **multivariate**

- boxes, 9
- bubbleFIN, 16
- cor.sign, 41
- CorCompare, 42
- CorGroups, 43
- factanal.fit.principal, 48
- loadplot, 52
- pfa, 62
- plotmvoutlier, 66
- polys, 70
- rg.mva, 82
- rg.mvalloc, 84
- rg.robmva, 86
- rg.wtdsums, 88
- suns, 97
- ternary, 100

## \* **robust**

- rg.mvalloc, 84
- rg.robmva, 86
- RobCor.plot, 89

## \* **smooth**

- SmoothLegend, 95

- \* **spatial**
  - polygrid, 69
- \* **univar**
  - boxplotlog, 12
  - boxplotperc, 14
  - cor.sign, 41
  - CorCompare, 42
  - do.ellipses, 44
  - edaplot, 45
  - edaplotlog, 47
  - plotellipse, 65
  - plotuniout, 67
  - rg.boxplot, 81
  - RobCor.plot, 89
  - varcomp, 105
- arw, 3, 67, 68
- AuNEW, 4
- AuOLD, 5
- bhorizon, 5
- bordersKola, 8
- box, 10
- boxes, 9
- boxplot, 46, 48
- boxplotlegend, 11
- boxplotlog, 12, 15, 48
- boxplotperc, 14
- bubbleFIN, 16
- CHorANADUP, 17
- CHorFieldDUP, 23
- chorizon, 30
- CHorSTANDARD, 34
- concare, 37, 40
- concareExampleKola, 38, 39
- cor.sign, 41
- cor.test, 41
- CorCompare, 42
- CorGroups, 43
- covMcd, 67, 68
- do.ellipses, 44
- edaplot, 45
- edaplotlog, 46, 47
- expand.grid, 69
- factanal.fit.principal, 48
- hist, 46, 48
- kola.background, 49
- KrigeLegend, 50
- loadplot, 52
- monch, 53
- moss, 54
- nizap, 57
- Northarrow, 57
- ohorizon, 58
- over, 69
- par, 74, 75
- pfa, 62
- plot, 46, 48, 74
- plot.default, 10, 74
- plotbg, 63, 67
- plotelement, 64
- plotellipse, 65
- plotmvoutlier, 66
- plotuniout, 67
- points, 46, 48
- polygrid, 69
- polys, 70
- ppplot.das, 72
- qqplot.das, 40, 73
- qqplot.das, 74
- res.eyefit.As\_C, 76
- res.eyefit.As\_C\_m, 76
- res.eyefit.AuNEW, 77
- res.eyefit.Ca\_C, 78
- res.eyefit.Ca\_0, 78
- res.eyefit.Hg\_0, 79
- res.eyefit.Pb\_01, 80
- res.eyefit.Pb\_02, 80
- rg.boxplot, 81
- rg.mva, 82
- rg.mvalloc, 84
- rg.remove.na, 85
- rg.robmva, 86
- rg.wtdsums, 88
- RobCor.plot, 89
- roundpretty, 90, 91
- roundpretty.sub, 90, 91

scalebar, [92](#)  
scatter3dPETER, [93](#)  
SmoothLegend, [95](#)  
SpatialPoints, [69](#)  
suns, [97](#)  
SymbLegend, [98](#)  
  
ternary, [100](#)  
timetrend, [101](#)  
topsoil, [103](#)  
  
varcomp, [105](#)