The mvoutlier Package

May 16, 2007

Version 1.3

Date 2007-05-09

Title Multivariate outlier detection based on robust methods

Author Moritz Gschwandtner <e0125439@student.tuwien.ac.at> and Peter Filzmoser <P.Filzmoser@tuwien.ac.at>

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

Depends R (>= 1.9.0), robustbase, stats

Description This packages was made for multivariate outlier detection.

License GPL version 2 or newer

URL http://www.statistik.tuwien.ac.at/public/filz/

R topics documented:

aq.plot ........................................... 2
arw .................................................. 3
bhorizon ......................................... 4
bss.background ................................. 6
bssbot ......................................... 7
bsstop ......................................... 9
chisq.plot ................................... 11
chorizon .................................... 12
color.plot .................................. 16
cor.plot .................................... 17
dd.plot ..................................... 18
humus ....................................... 19
kola.background .............................. 21
map.plot .................................... 22
moss ......................................... 23
pbb ......................................... 25
pcout ...................................... 26
Description

The function `aq.plot` plots the ordered squared robust Mahalanobis distances of the observations against the empirical distribution function of the $MD_i^2$. In addition the distribution function of $\text{chisq}_p$ is plotted as well as two vertical lines corresponding to the chisq-quantile specified in the argument list (default is 0.975) and the so-called adjusted quantile. Three additional graphics are created (the first showing the data, the second showing the outliers detected by the specified quantile of the $\text{chisq}_p$ distribution and the third showing these detected outliers by the adjusted quantile).

Usage

```r
aq.plot(x, delta=qchisq(0.975, df=ncol(x)), quan=1/2, alpha=0.025)
```

Arguments

- `x` : matrix or `data.frame` containing the data; has to be at least two-dimensional
- `delta` : quantile of the chi-squared distribution with `ncol(x)` degrees of freedom. This quantile appears as cyan-colored vertical line in the plot.
- `quan` : proportion of observations which are used for MCD estimations; has to be between 0.5 and 1, default ist 0.5
- `alpha` : Maximum thresholding proportion (optional scalar, default: $\alpha = 0.025$)

Details

The function `aq.plot` plots the ordered squared robust Mahalanobis distances of the observations against the empirical distribution function of the $MD_i^2$. The distance calculations are based on the MCD estimator.

For outlier detection two different methods are used. The first one marks observations as outliers if they exceed a certain quantile of the chi-squared distribution. The second is an adaptive procedure searching for outliers specifically in the tails of the distribution, beginning at a certain chisq-quantile (see Filzmoser et al., 2005).

The function behaves differently depending on the dimension of the data. If the data is more than two-dimensional the data are projected on the first two robust principal components.

Value

- `outliers` : boolean vector of outliers
Author(s)
Moritz Gschwandtner <e0125439@student.tuwien.ac.at>
Peter Filzmoser <P.Filzmoser@tuwien.ac.at> http://www.statistik.tuwien.ac.at/public/filz/

References

Examples

```r
# create data:
x <- cbind(rnorm(100), rnorm(100), rnorm(100))
y <- cbind(rnorm(10, 5, 1), rnorm(10, 5, 1), rnorm(10, 5, 1))
z <- rbind(x, y)
# execute:
aq.plot(z, alpha=0.1)
```

arw Adaptive reweighted estimator for multivariate location and scatter

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

```r
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://www.statistik.tuwien.ac.at/public/filz/

References


Examples

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

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 48 variables.

ID  a numeric vector
XCOO a numeric vector
YCOO a numeric vector
Ag  a numeric vector
Al  a numeric vector
Al_XRF a numeric vector
As  a numeric vector
Ba  a numeric vector
Be  a numeric vector
Bi  a numeric vector
Ca  a numeric vector
Ca_XRF a numeric vector
Cd  a numeric vector
Co  a numeric vector
Cr  a numeric vector
Cu  a numeric vector
EC  a numeric vector
Fe  a numeric vector
Fe_XRF a numeric vector
K   a numeric vector
K_XRF a numeric vector
LOI a numeric vector
La  a numeric vector
Li  a numeric vector
Mg  a numeric vector
Mg_XRF a numeric vector
Mn  a numeric vector
Mn_XRF a numeric vector
Mo  a numeric vector
Na  a numeric vector
Na_XRF a numeric vector
Ni  a numeric vector
P   a numeric vector
P_XRF a numeric vector
Pb  a numeric vector
S   a numeric vector
Sc  a numeric vector
Se  a numeric vector
Si  a numeric vector
Si_XRF a numeric vector
Sr  a numeric vector
Te  a numeric vector
Th  a numeric vector
Ti  a numeric vector
Ti_XRF a numeric vector
V   a numeric vector
Y   a numeric vector
Zn  a numeric vector
Source

References

Examples
```
data(bhorizon)
# classical versus robust correlation
cor.plot(log(bhorizon[,"Al"]), log(bhorizon[,"Na"]))
```

Description
Coordinates of the BSS data background map

Usage
```
data(bss.background)
```

Format
A data frame with 6093 observations on the following 2 variables.

V1 a numeric vector with the x-coordinates
V2 a numeric vector with the y-coordinates

Details
Is used by pbb()

Source
BSS project

References
Examples

```r
data(bss.background)
pbb()
```

---

**Bottom Layer of the BSS Data**

**Description**

The BSS data were collected in agrigultural soils from Northern Europe. from an area of about 1,800,000 km². 769 samples on an irregular grid were taken in two different layers, the top layer (0-20cm) and the bottom layer. This dataset contains the bottom layer of the BSS data. It has 46 variables, including x and y coordinates.

**Usage**

```r
data(bssbot)
```

**Format**

A data frame with 768 observations on the following 46 variables.

- **ID** a numeric vector
- **CNo** a numeric vector
- **XCOO** x coordinates: a numeric vector
- **YCOO** y coordinates: a numeric vector
- **SiO2_B** a numeric vector
- **TiO2_B** a numeric vector
- **Al2O3_B** a numeric vector
- **Fe2O3_B** a numeric vector
- **MnO_B** a numeric vector
- **MgO_B** a numeric vector
- **CaO_B** a numeric vector
- **Na2O_B** a numeric vector
- **K2O_B** a numeric vector
- **P2O5_B** a numeric vector
- **SO3_B** a numeric vector
- **Cl_B** a numeric vector
- **F_B** a numeric vector
- **LOI_B** a numeric vector
- **As_B** a numeric vector
**Source**

BSS Project in Northern Europe

**References**


**Examples**

data(bssbot)
# classical versus robust correlation
cor.plot(log(bssbot[, "Al2O3_B"]), log(bssbot[, "Na2O_B"]))
Top Layer of the BSS Data

Description
The BSS data were collected in agricultural soils from Northern Europe from an area of about 1,800,000 km². 769 samples on an irregular grid were taken in two different layers, the top layer (0-20 cm) and the bottom layer. This dataset contains the top layer of the BSS data. It has 46 variables, including x and y coordinates.

Usage

data(bsstop)

Format
A data frame with 768 observations on the following 46 variables.

- ID a numeric vector
- CNo a numeric vector
- XCOO x coordinates: a numeric vector
- YCOO y coordinates: a numeric vector
- SiO2_T a numeric vector
- TiO2_T a numeric vector
- Al2O3_T a numeric vector
- Fe2O3_T a numeric vector
- MnO_T a numeric vector
- MgO_T a numeric vector
- CaO_T a numeric vector
- Na2O_T a numeric vector
- K2O_T a numeric vector
- P2O5_T a numeric vector
- SO3_T a numeric vector
- Cl_T a numeric vector
- F_T a numeric vector
- LOI_T a numeric vector
- As_T a numeric vector
- Ba_T a numeric vector
- Bi_T a numeric vector
- Ce_T a numeric vector
bsstop

Co_T a numeric vector
Cr_T a numeric vector
Cs_T a numeric vector
Cu_T a numeric vector
Ga_T a numeric vector
Hf_T a numeric vector
La_T a numeric vector
Mo_T a numeric vector
Nb_T a numeric vector
Ni_T a numeric vector
Pb_T a numeric vector
Rb_T a numeric vector
Sb_T a numeric vector
Sc_T a numeric vector
Sn_T a numeric vector
Sr_T a numeric vector
Ta_T a numeric vector
Th_T a numeric vector
U_T a numeric vector
V_T a numeric vector
W_T a numeric vector
Y_T a numeric vector
Zn_T a numeric vector
Zr_T a numeric vector

Source
BSS Project in Northern Europe

References

Examples
data(bsstop)
# classical versus robust correlation
cor.plot(log(bsstop[, "Al2O3_T"]), log(bsstop[, "Na2O_T"]))
chisq.plot

**Chi-Square Plot**

**Description**

The function chisq.plot plots the ordered robust mahalanobis distances of the data against the quantiles of the Chi-squared distribution. By user interaction this plotting is iterated each time leaving out the observation with the greatest distance.

**Usage**

```r
chisq.plot(x, quan=1/2, ask=TRUE, ...)
```

**Arguments**

- `x`: matrix or data.frame containing the data
- `quan`: amount of observations which are used for mcd estimations. has to be between 0.5 and 1, default is 0.5
- `ask`: logical. specifies whether user interaction is allowed or not. default is TRUE
- `...`: additional graphical parameters

**Details**

The function chisq.plot plots the ordered robust mahalanobis distances of the data against the quantiles of the Chi-squared distribution. If the data is normal distributed these values should approximately correspond to each other, so outliers can be detected visually. By user interaction this procedure is repeated, each time leaving out the observation with the greatest distance (the number of the observation is printed on the console). This method can be seen as an iteratively deletion of outliers until a straight line appears.

**Value**

- `outliers`: Each time an observation is dropped, its index is added to a vector named outliers, which is accessible within the global environment after cancellation of the function.

**Author(s)**

Moritz Gschwandtner &lt;e0125439@student.tuwien.ac.at&gt;
Peter Filzmoser &lt;P.Filzmoser@tuwien.ac.at&gt; [http://www.statistik.tuwien.ac.at/public/filz/](http://www.statistik.tuwien.ac.at/public/filz/)

**References**

Examples

data(humus)
chisq.plot(log(humus[,c("Co","Cu","Ni")]))
# The identified outliers are in object "outliers"

chorizon  

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(chorizon)

Format

A data frame with 606 observations on the following 110 variables.

- **ID**  a numeric vector
- **XCOO**  a numeric vector
- **YCOO**  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_IC**  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_IC  a numeric vector
Co  a numeric vector
Co_INAA  a numeric vector
EC  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_IC  a numeric vector
Fe  a numeric vector
Fe_INAA  a numeric vector
Fe2O3  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
K2O  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
wt_INAA  a numeric vector
Mg  a numeric vector
MgO  a numeric vector
Mn  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
Na2O  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
P2O5 a numeric vector
Pb a numeric vector
pH a numeric vector
PO4_IC 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_IC a numeric vector
Sr a numeric vector
Sr_INAA a numeric vector
SUM_XRF 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
chordon

Zn  a numeric vector
Zn_INAA  a numeric vector
ELEV  a numeric vector
COUN  a numeric vector
ASP  a numeric vector
TOPC  a numeric vector
LITO  a numeric vector
Al_XRF  a numeric vector
Ca_XRF  a numeric vector
Fe_XRF  a numeric vector
K_XRF  a numeric vector
Mg_XRF  a numeric vector
Mn_XRF  a numeric vector
Na_XRF  a numeric vector
P_XRF  a numeric vector
Si_XRF  a numeric vector
Ti_XRF  a numeric vector

Source


References


Examples

data(chordon)
# classical versus robust correlation
cor.plot(log(chordon[,"Al"]), log(chordon[,"Na"]))
Description
The function color.plot plots the (two-dimensional) data using different symbols according to the robust mahalanobis distance based on the mcd estimator with adjustment and using different colors according to the euclidean distances of the observations.

Usage

```
color.plot(x, quan=1/2, alpha=0.025, ...)```

Arguments

- **x**: two dimensional matrix or data.frame containing the data.
- **quan**: amount of observations which are used for mcd estimations. has to be between 0.5 and 1, default ist 0.5
- **alpha**: amount of observations used for calculating the adjusted quantile (see function arw).
- **...**: additional graphical parameters

Details
The function color.plot plots the (two-dimensional) data using different symbols (see function symbol.plot) according to the robust mahalanobis distance based on the mcd estimator with adjustment and using different colors according to the euclidean distances of the observations. Blue is typical for a little distance, whereas red is the opposite. In addition four ellipsoids are drawn, on which mahalanobis distances are constant. These constant values correspond to the 25%, 50%, 75% and adjusted quantiles (see function arw) of the chi-square distribution (see Filzmoser et al., 2005).

Value

- **outliers**: boolean vector of outliers
- **md**: robust mahalanobis distances of the data
- **euclidean**: euclidean distances of the observations according to the minimum of the data.

Author(s)
Moritz Gschwandtner <e0125439@student.tuwien.ac.at>
Peter Filzmoser <P.Filzmoser@tuwien.ac.at> http://www.statistik.tuwien.ac.at/public/filz/

References
cor.plot

See Also
symbol.plot.dd.plot.arw

Examples

```r
# create data:
x <- cbind(rnorm(100), rnorm(100))
y <- cbind(rnorm(10, 5, 1), rnorm(10, 5, 1))
z <- rbind(x,y)
# execute:
color.plot(z, quan=0.75)
```

cor.plot  Correlation Plot: robust versus classical bivariate correlation

Description

The function cor.plot plots the (two-dimensional) data and adds two correlation ellipsoids, based on classical and robust estimation of location and scatter. Robust estimation can be thought of as estimating the mean and covariance of the 'good' part of the data.

Usage

cor.plot(x, y, quan=1/2, alpha=0.025, ...)

Arguments

- `x`: vector to be plotted against y and of which the correlation with y is calculated.
- `y`: vector to be plotted against x and of which the correlation with x is calculated.
- `quan`: amount of observations which are used for mcd estimations. has to be between 0.5 and 1, default ist 0.5
- `alpha`: Determines the size of the ellipsoids. An observation will be outside of the ellipsoid if its mahalanobis distance exceeds the 1-alpha quantile of the chi-squared distribution.
- `...`: additional graphical parameters

Value

- `cor.cla`: correlation between x and y based on classical estimation of location and scatter
- `cor.rob`: correlation between x and y based on robust estimation of location and scatter

Author(s)

Moritz Gschwandtner <e0125439@student.tuwien.ac.at>
Peter Filzmoser <P.Filzmoser@tuwien.ac.at> http://www.statistik.tuwien.ac.at/public/filz/
See Also

covMcd

Examples

# create data:
x <- cbind(rnorm(100), rnorm(100))
y <- cbind(rnorm(10, 3, 1), rnorm(10, 3, 1))
z <- rbind(x,y)
# execute:
cor.plot(z[,1], z[,2])

dd.plot  Distance-Distance Plot

Description

The function dd.plot plots the classical mahalanobis distance of the data against the robust mahalanobis distance based on the mcd estimator. Different symbols (see function symbol.plot) and colours (see function color.plot) are used depending on the mahalanobis and euclidean distance of the observations (see Filzmoser et al., 2005).

Usage

dd.plot(x, quan=1/2, alpha=0.025, ...)

Arguments

x                      matrix or data frame containing the data
quan                   amount of observations which are used for mcd estimations. has to be between 0.5 and 1, default is 0.5
alpha                  amount of observations used for calculating the adjusted quantile (see function arw).
...                    additional graphical parameters

Value

outliers       boolean vector of outliers
md.cla         mahalanobis distances of the observations based on classical estimators of location and scatter.
md.rob         mahalanobis distances of the observations based on robust estimators of location and scatter (mcd).

Author(s)

Moritz Gschwandtner <e0125439@student.tuwien.ac.at>
Peter Filzmoser <P.Filzmoser@tuwien.ac.at> http://www.statistik.tuwien.ac.at/public/filz/
**References**


**See Also**

`symbol.plot`, `color.plot`, `arw`, `covPlot`

**Examples**

```r
# create data:
x <- cbind(rnorm(100), rnorm(100))
y <- cbind(rnorm(10, 3, 1), rnorm(10, 3, 1))
z <- rbind(x,y)
# execute:
dd.plot(z)
#
# Identify multivariate outliers for Co-Cu-Ni in humus layer of Kola data:
data(humus)
dd.plot(log(humus[,c("Co","Cu","Ni")]))
```

---

**Humus Layer (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 humus layer.

**Usage**

data(humus)

**Format**

A data frame with 617 observations on the following 44 variables.

- **ID** a numeric vector
- **XCOO** a numeric vector
- **YCOO** 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
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
Rb  a numeric vector
S   a numeric vector
Sb  a numeric vector
Sc  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
C   a numeric vector
H   a numeric vector
N   a numeric vector
LOI a numeric vector
pH  a numeric vector
Cond a numeric vector
kola.background

Source


References


Examples

data(humus)
# classical versus robust correlation:
cor.plot(log(humus[,"Al"]), log(humus[,"Na"]))

Description

Coordinates of the kola.map background

Usage

data(kola.background)

Format


Details

Is used by map.plot()

Source

References


Examples

```r
example(map.plot)
```

---

**map.plot**  
*Plot Multivariate Outliers in a Map*

**Description**

The function `map.plot` creates a map using geographical (x,y)-coordinates. This is thought for spatially dependent data of which coordinates are available. Multivariate outliers are marked.

**Usage**

```r
map.plot(coord, data, quan=1/2, alpha=0.025, symb=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), ...)
```

**Arguments**

- `coord`: (x,y)-coordinates of the data
- `data`: matrix or data.frame containing the data.
- `quan`: amount of observations which are used for mcd estimations. has to be between 0.5 and 1, default ist 0.5
- `alpha`: amount of observations used for calculating the adjusted quantile (see function `arw`).
- `symb`: logical for plotting special symbols (see details).
- `plotmap`: logical for plotting the background map.
- `map`: see `plot.kola.background()`
- `which.map`: see `plot.kola.background()`
- `map.col`: see `plot.kola.background()`
- `map.lwd`: see `plot.kola.background()`
- ...: additional graphical parameters

**Details**

The function `map.plot` shows multivariate outliers in a map. If `symb=FALSE` (default), only two colors and no special symbols are used to mark multivariate outliers (the outliers are marked red). If `symb=TRUE` different symbols and colors are used. The symbols (cross means big value, circle means little value) are selected according to the robust mahalanobis distance based on the adjusted mcd estimator (see function `symbol.plot`) Different colors (red means big value, blue means little value) according to the euclidean distances of the observations (see function `color.plot`) are used. For details see Filzmoser et al. (2005).
### Value

- **outliers**: boolean vector of outliers
- **md**: robust mahalanobis distances of the data
- **euclidean**: (only if symb=TRUE) euclidean distances of the observations according to the minimum of the data.

### Author(s)

Moritz Gschwandtner &lt;e0125439@student.tuwien.ac.at&gt;  
Peter Filzmoser &lt;P.Filzmoser@tuwien.ac.at&gt;  
[http://www.statistik.tuwien.ac.at/public/filz/](http://www.statistik.tuwien.ac.at/public/filz/)

### References


### See Also

`symbol.plot`, `color.plot`, `arw`

### Examples

```r
data(humus) # Load humus data
x <- humus[,c("XCOO","YCOO")]
myhumus <- log(humus[, c("As", "Cd", "Co", "Cu", "Mg", "Pb", "Zn")])
map.plot(x, myhumus, symb=TRUE)
```

---

### 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

```r
data(moss)
```
Format

A data frame with 598 observations on the following 34 variables.

- **ID** a numeric vector
- **XCOO** a numeric vector
- **YCOO** a numeric vector
- **Ag** a numeric vector
- **Al** a numeric vector
- **As** a numeric vector
- **B** a numeric vector
- **Ba** 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
- **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
- **Rb** a numeric vector
- **S** a numeric vector
- **Sb** 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
- **Zn** a numeric vector
Source


References


Examples

data(moss)
# classical versus robust correlation:
cor.plot(log(moss[,"Al"])), log(moss[,"Na"]))

pbb

BSS background Plot

Description

Plots the BSS background map

Usage

pbb(map = "bss.background", add.plot = FALSE, ...)

Arguments

map List of coordinates. For the correct format see also help(kola.background)
add.plot logical. If true background is added to an existing plot
... additional plot parameters, see help(par)

Details

The list of coordinates is plotted as a polygon line.

Value

The plot is produced on the graphical device.

Author(s)

Peter Filzmoser <(P.Filzmoser@tuwien.ac.at)> http://www.statistik.tuwien.ac.at/public/filz/
References


See Also

See also pkb

Examples

```r
data(bss.background)
data(bsstop)
plot(bsstop$XCOO,bsstop$YCOO,col="red",pch=3)
pbb(add=TRUE)
```

pcout  

**PCOut Method for Outlier Identification in High Dimensions**

Description

Fast algorithm for identifying multivariate outliers in high-dimensional and/or large datasets, using the algorithm of Filzmoser, Maronna, and Werner (CSDA, 2007).

Usage

```r
pcout(x, makeplot = FALSE, explvar = 0.99, crit.M1 = 1/3, crit.c1 = 2.5, crit.M2 = 0.25, crit.c2 = 0.99, cs = 0.25, ...)  # Arguments
```

Arguments

- `x`: a numeric matrix or data frame which provides the data for outlier detection
- `makeplot`: a logical value indicating whether a diagnostic plot should be generated (default to FALSE)
- `explvar`: a numeric value between 0 and 1 indicating how much variance should be covered by the robust PCs (default to 0.99)
- `crit.M1`: a numeric value between 0 and 1 indicating the quantile to be used as lower boundary for location outlier detection (default to 1/3)
- `crit.c1`: a positive numeric value used for determining the upper boundary for location outlier detection (default to 2.5)
- `crit.M2`: a numeric value between 0 and 1 indicating the quantile to be used as lower boundary for scatter outlier detection (default to 1/4)
- `crit.c2`: a numeric value between 0 and 1 indicating the quantile to be used as upper boundary for scatter outlier detection (default to 0.99)
- `cs`: a numeric value indicating the scaling constant for combined location and scatter weights (default to 0.25)
outbound

A numeric value between 0 and 1 indicating the outlier boundary for defining values as final outliers (default to 0.25)

... additional plot parameters, see help(par)

Details

Based on the robustly sphered data, semi-robust principal components are computed which are needed for determining distances for each observation. Separate weights for location and scatter outliers are computed based on these distances. The combined weights are used for outlier identification.

Value

wfinal01 0/1 vector with final weights for each observation; weight 0 indicates potential multivariate outliers.
wfinal numeric vector with final weights for each observation; small values indicate potential multivariate outliers.
wloc numeric vector with weights for each observation; small values indicate potential location outliers.
wscat numeric vector with weights for each observation; small values indicate potential scatter outliers.
x.dist1 numeric vector with distances for location outlier detection.
x.dist2 numeric vector with distances for scatter outlier detection.
M1 upper boundary for assigning weight 1 in location outlier detection.
const1 lower boundary for assigning weight 0 in location outlier detection.
M2 upper boundary for assigning weight 1 in scatter outlier detection.
const2 lower boundary for assigning weight 0 in scatter outlier detection.

Author(s)

Peter Filzmoser <(P.Filzmoser@tuwien.ac.at)>

References


See Also

sign1, sign2
Examples

```r
# geochemical data from northern Europe
data(bsstop)
x = bsstop[, 5:14]
# identify multivariate outliers
x.out = pcout(x, makeplot = FALSE)
# visualize multivariate outliers in the map
op <- par(mfrow = c(1, 2))
data(bss.background)
pbb(asp = 1)
points(bsstop$XCOO, bsstop$YCOO, pch = 16, col = x.out$wfinal01 + 2)
title("Outlier detection based on pcout")
legend("topleft", legend = c("potential outliers", "regular observations"), pch = 16, col = c(2, 3))

# compare with outlier detection based on MCD:
require(robustbase)
x.mcd = covMcd(x)
pbb(asp = 1)
points(bsstop$XCOO, bsstop$YCOO, pch = 16, col = x.mcd$mcd.wt + 2)
title("Outlier detection based on MCD")
legend("topleft", legend = c("potential outliers", "regular observations"), pch = 16, col = c(2, 3))
par(op)
```

Description

Plots the Kolamap background

Usage

```r
pkb(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 help(kola.background)
- `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 help(par)

Details

Is used by map.plot()
Author(s)

Peter Filzmoser <P.Filzmoser@tuwien.ac.at> http://www.statistik.tuwien.ac.at/public/filz/

References


Examples

eexample(map.plot)

---

sign1  

Sign Method for Outlier Identification in High Dimensions - Simple Version

Description

Fast algorithm for identifying multivariate outliers in high-dimensional and/or large datasets, using spatial signs, see Filzmoser, Maronna, and Werner (CSDA, 2007). The computation of the distances is based on Mahalanobis distances.

Usage

sign1(x, makeplot = FALSE, qcrit = 0.975, ...)

Arguments

x  
a numeric matrix or data frame which provides the data for outlier detection

makeplot  
a logical value indicating whether a diagnostic plot should be generated (default to FALSE)

qcrit  
a numeric value between 0 and 1 indicating the quantile to be used as critical value for outlier detection (default to 0.975)

...  
additional plot parameters, see help(par)

Details

Based on the robustly sphered and normed data, robust principal components are computed. These are used for computing the covariance matrix which is the basis for Mahalanobis distances. A critical value from the chi-square distribution is then used as outlier cutoff.
Value

- `wfinal01`: 0/1 vector with final weights for each observation; weight 0 indicates potential multivariate outliers.
- `x.dist`: numeric vector with distances used for outlier detection.
- `const`: outlier cutoff value.

Author(s)

Peter Filzmoser <(P.Filzmoser@tuwien.ac.at)>

References


See Also

- `pcout`, `sign2`

Examples

```r
# geochemical data from northern Europe
data(bsstop)
x=bsstop[,5:14]
# identify multivariate outliers
x.out=sign1(x,makeplot=FALSE)
# visualize multivariate outliers in the map
op <- par(mfrow=c(1,2))
data(bsstop.background)
pbb(asp=1)
points(bsstop$XCOO,bsstop$YCOO,pch=16,col=x.out$wfinal01+2)
title("Outlier detection based on signout")
legend("topleft",legend=c("potential outliers","regular observations"),pch=16,col=c(2,3))

# compare with outlier detection based on MCD:
require(robustbase)
x.mcd=covMcd(x)
pbb(asp=1)
points(bsstop$XCOO,bsstop$YCOO,pch=16,col=x.mcd$mcd.wt+2)
title("Outlier detection based on MCD")
legend("topleft",legend=c("potential outliers","regular observations"),pch=16,col=c(2,3))
par(op)
```
Description

Fast algorithm for identifying multivariate outliers in high-dimensional and/or large datasets, using spatial signs, see Filzmoser, Maronna, and Werner (CSDA, 2007). The computation of the distances is based on principal components.

Usage

`sign2(x, makeplot = FALSE, explvar = 0.99, qcrit = 0.975, ...)`

Arguments

- `x` a numeric matrix or data frame which provides the data for outlier detection
- `makeplot` a logical value indicating whether a diagnostic plot should be generated (default to FALSE)
- `explvar` a numeric value between 0 and 1 indicating how much variance should be covered by the robust PCs (default to 0.99)
- `qcrit` a numeric value between 0 and 1 indicating the quantile to be used as critical value for outlier detection (default to 0.975)
- `...` additional plot parameters, see help(par)

Details

Based on the robustly sphered and normed data, robust principal components are computed which are needed for determining distances for each observation. The distances are transformed to approach chi-square distribution, and a critical value is then used as outlier cutoff.

Value

- `wfinal01` 0/1 vector with final weights for each observation; weight 0 indicates potential multivariate outliers.
- `x.dist` numeric vector with distances used for outlier detection.
- `const` outlier cutoff value.

Author(s)

Peter Filzmoser <(P.Filzmoser@tuwien.ac.at)>
References


See Also

`pcout, sign1`

Examples

```r
# geochemical data from northern Europe
data(bsstop)
x=bsstop[,5:14]
# identify multivariate outliers
x.out=sign2(x,makeplot=FALSE)
# visualize multivariate outliers in the map
op <- par(mfrow=c(1,2))
data(bss.background)
pbb(asp=1)
points(bsstop$XCOO,bsstop$YCOO,pch=16,col=x.out$wfinal01+2)
title("Outlier detection based on signout")
legend("topleft",legend=c("potential outliers","regular observations"),pch=16,col=c(2,3))

# compare with outlier detection based on MCD:
require(robustbase)
x.mcd=covMcd(x)
pbb(asp=1)
points(bsstop$XCOO,bsstop$YCOO,pch=16,col=x.mcd$mcd.wt+2)
title("Outlier detection based on MCD")
legend("topleft",legend=c("potential outliers","regular observations"),pch=16,col=c(2,3))
par(op)
```

symbol.plot  Symbol Plot

Description

The function `symbol.plot` plots the (two-dimensional) data using different symbols according to the robust mahalanobis distance based on the mcd estimator with adjustment.

Usage

```r
symbol.plot(x, quan=1/2, alpha=0.025, ...)
```
symbol.plot

Arguments

x  two dimensional matrix or data.frame containing the data.
quan amount of observations which are used for mcd estimations. has to be between 0.5 and 1, default ist 0.5
alpha amount of observations used for calculating the adjusted quantile (see function arw).
... additional graphical parameters

Details

The function symbol.plot plots the (two-dimensional) data using different symbols. In addition a legend and four ellipsoids are drawn, on which mahalanobis distances are constant. As the legend shows, these constant values correspond to the 25%, 50%, 75% and adjusted (see function arw) quantiles of the chi-square distribution.

Value

outliers boolean vector of outliers
md robust mahalanobis distances of the data

Author(s)

Moritz Gschwandtner \(<\text{e0125439@student.tuwien.ac.at}>\)
Peter Filzmoser \(<\text{P.Filzmoser@tuwien.ac.at}>\) \text{http://www.statistik.tuwien.ac.at/public/filz/}

References


See Also

dd.plot, color.plot, arw

Examples

# create data:
x <- cbind(rnorm(100), rnorm(100))
y <- cbind(rnorm(10, 5, 1), rnorm(10, 5, 1))
z <- rbind(x, y)
# execute:
symbol.plot(z, quan=0.75)
uni.plot

Univariate Presentation of Multivariate Outliers

Description

The function uni.plot plots each variable of x parallel in a one-dimensional scatter plot and in addition marks multivariate outliers.

Usage

uni.plot(x, symb=FALSE, quan=1/2, alpha=0.025, ...)

Arguments

x matrix or data.frame containing the data.

symb logical. if FALSE, only two colors and no special symbols are used. outliers are marked red. if TRUE different symbols (cross means big value, circle means little value) according to the robust mahalanobis distance based on the mcd estimator and different colors (red means big value, blue means little value) according to the euclidean distances of the observations are used.

quan amount of observations which are used for mcd estimations. has to be between 0.5 and 1, default ist 0.5

alpha amount of observations used for calculating the adjusted quantile (see function arw).

... additional graphical parameters

Details

The function uni.plot shows the multivariate outliers in the single variables by one-dimensional scatter plots. If symb=FALSE (default), only two colors and no special symbols are used to mark multivariate outliers (the outliers are marked red). If symb=TRUE different symbols and colors are used. The symbols (cross means big value, circle means little value) are selected according to the robust mahalanobis distance based on the adjusted mcd estimator (see function symbol.plot) Different colors (red means big value, blue means little value) according to the euclidean distances of the observations (see function color.plot) are used. For details see Filzmoser et al. (2005).

Value

outliers boolean vector of outliers

md robust multivariate mahalanobis distances of the data

euclidean (only if symb=TRUE) multivariate euclidean distances of the observations according to the minimum of the data.
Author(s)

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

References


See Also

map.plot, symbol.plot, color.plot, arw

Examples

data(swiss)
uni.plot(swiss)
#
# Geostatistical data:
data(humus) # Load humus data
uni.plot(log(humus[, c("As", "Cd", "Co", "Cu", "Mg", "Pb", "Zn")]), symb=TRUE)
Index

*Topic datasets
  bhorizon, 4
  bss.background, 6
  bssbot, 7
  bsstop, 9
  chorizon, 12
  humus, 19
  kola.background, 21
  moss, 23
  pbb, 25
  pkb, 28

*Topic dplot
  aq.plot, 1
  arw, 3
  chisq.plot, 11
  color.plot, 16
  cor.plot, 17
  dd.plot, 18
  map.plot, 22
  symbol.plot, 32
  uni.plot, 34

*Topic multivariate
  pcout, 26
  sign1, 29
  sign2, 31

*Topic robust
  pcout, 26
  sign1, 29
  sign2, 31

aq.plot, 1
arw, 3, 17, 19, 23, 33, 35

bhorizon, 4
bss.background, 6
bssbot, 7
bsstop, 9

chisq.plot, 11
chorizon, 12

color.plot, 16, 19, 23, 33, 35
cor.plot, 17
covMcd, 18
covPlot, 19
dd.plot, 17, 18, 33
humus, 19
kola.background, 21
map.plot, 22, 35
moss, 23
pbb, 25
pcout, 26, 30, 32
pkb, 26, 28
sign1, 27, 29, 32
sign2, 27, 30, 31
symbol.plot, 17, 19, 23, 32, 35
uni.plot, 34

36