Package ‘robcor’

October 14, 2022

Type Package
Title Robust Correlations
Version 0.1-6.1
Date 2014-01-06
Author Paul Smirnov
Maintainer Paul Smirnov <s.paul@mail.ru>
Description Robust pairwise correlations based on estimates of scale,
          particularly on  ```FastQn``` one-step M-estimate.
Suggests MASS, robustbase, sfsmisc
Depends R (>= 2.10.0), stats
License GPL (>= 2)
NeedsCompilation no
Repository CRAN
Date/Publication 2022-06-27 05:20:51 UTC

R topics documented:

  FastQn ................................................................. 2
  psdcor ............................................................... 3
  robacf ............................................................... 4
  robacf ............................................................... 5
  robcor ............................................................... 6

Index 8
**FastQn**  
*Robust, Efficient and Fast Scale Estimate*

---

**Description**

Compute the robust scale estimator FastQn, an efficient alternative to the MAD, a fast alternative to the Qn.

**Usage**

```
FastQn(x, center = median(x), scale = mad(x, center))
```

```
fqn(x, center = median(x), scale = mad(x, center))
```

```
s_FastQn(x, mu.too = FALSE, center = median(x), ...)
```

**Arguments**

- `x` numeric vector of observations.
- `center` optionally, the center: defaults to the median.
- `scale` optionally, the basic scale: defaults to the median absolute deviation.
- `mu.too` logical indicating if the center should also be returned for `s_FastQn()`.
- `...` potentially further arguments for `s_FastQn()` passed to `FastQn()`.

**Details**

This function computes one-step M-estimate of scale based on provided robust estimate (defaults to the MAD). It gives 50% breakdown point and Gaussian efficiency about 80%.

The `fqn` function is a shorter alias, like `sd` and `mad`.

**Value**

`FastQn()` returns a number, the FastQn robust scale estimator.

`s_FastQn(x, mu.too=TRUE)` returns a length-2 vector with location and scale; this is typically only useful for `covOGK(*, sigmamu = s_FastQn)` or `robcor(*, scaler = s_FastQn)`.

**Author(s)**

Paul Smirnov <s.paul@mail.ru>

**References**

psdcor

See Also
mad, Qn.

Examples

```r
set.seed(153)
x <- sort(c(rnorm(80), rt(20, df = 1)))
s_FastQn(x, mu.too=TRUE)
FastQn(x)
```

psdcor  Positive Semidefinite Correlation Matrix Correction

Description
Correct pseudo-correlation matrices to make them positive semidefinite ones.

Usage

```r
psdcor(m, method = c("higham", "eigen"), ...)
```

Arguments

- `m`  pseudo-correlation matrix to correct.
- `method`  optionally, correction method to use.
- `...`  potentially further arguments for the particular correction method.

Details
This function applies transformation to the given matrix in order to make it positive semidefinite correlation matrix.

When method is higham, use Higham projections algorithm via `nearPD()` function.
When method is eigen, use direct eigenvalues correction via `posdefify()` function.

Value
The corrected matrix.

Author(s)
Paul Smirnov <s.paul@mail.ru>
robacf  Robust Autocovariance and Autocorrelation Function Estimation

Description
Compute (and by default plot) an estimate of the autocovariance or autocorrelation function.

Usage
robacf(x, lag.max = NULL, type = c("correlation", "covariance"), plot = TRUE, scaler = "s_FastQn", ...)

Arguments
- x: a univariate numeric time series object or a numeric vector.
- lag.max: maximum lag at which to calculate the acf. Default is $10 \log_{10}(N)$ where $N$ is the number of observations. Will be automatically limited to one less than the number of observations in the series.
- type: character string giving the type of acf to be computed. Allowed values are "correlation" (the default) or "covariance".
- plot: logical. If TRUE (the default) the acf is plotted.
- scaler: location-scale estimator to use in the algorithm. By default, s_FastQn() is used.
- ...: further arguments to be passed to plot.acf.

Details
This function is a robust replacement for acf().
Note, that implementation and documentation is not finished/polished yet.

Value
A list of class "acf". For description of elements see acf().

Note
WORK-IN-PROGRESS status.

Author(s)
Paul Smirnov <s.paul@mail.ru>

References
robard  Robust Fit Autoregressive Models to Time Series

Description
Fit an autoregressive time series model to the data using robust algorithms.

Usage
robard(x, order = 2, scaler = "s_FastQn")

Arguments
x          a univariate time series.
order      an order of model to fit.
scaler     location-scale estimator to use in the algorithm. By default, s_FastQn() is used.

Details
This function is a robust replacement for ar().
Note, that implementation and documentation is not finished/polished yet.

Value
A list of class "ar". For description of elements see ar().

Note
WORK-IN-PROGRESS status.

Author(s)
Paul Smirnov <s.paul@mail.ru>

References

Examples
n <- 100
set.seed(361)
eps <- as.ts(rnorm(n))
x <- arima.sim(list(ar=c(1,-0.9)), n, innov=eps) # basic signal
z <- as.ts(rbinom(n, 1, 0.1) * rnorm(n, sd=10)) # noise
y <- x + z
spec.ar(robard(y, order=2))
Description

Compute a robust estimate of the correlation coefficient or correlation matrix via pairwise correlations.

Usage

robcor(x, y = NULL, method = c("ssd", "quadrant", "mcd"), partial = FALSE, post = "psdcor", scaler = "s_FastQn", regress = "lmrob")

Arguments

- **x**: a numeric vector, matrix or data frame.
- **y**: NULL (default) or a vector, matrix or data frame with compatible dimensions to x. The default is equivalent to y = x (but more efficient).
- **method**: a character string indicating which correlation coefficient is to be computed.
- **partial**: logical. Should a partial correlation algorithm be used?
- **post**: function to apply after the matrix is built or NULL. By default, positive semidefinite correction is applied (psdcor()).
- **scaler**: function to use as a location-scale estimator in "ssd" method. By default, s_FastQn() is used.
- **regress**: function to use as a regression estimator in partial correlations algorithm. By default, lmrob() is used.

Details

This function is a robust replacement for cor().

Note, that implementation and documentation is not finished/polished yet.

Value

Either a single correlation coefficient or a correlation matrix estimate.

Note

WORK-IN-PROGRESS status.

Author(s)

Paul Smirnov <s.paul@mail.ru>
References

Index

* array
  psdcor, 3
* multivariate
  robcor, 6
* robust
  robacf, 4
  robar, 5
  robcor, 6
* ts
  robacf, 4
  robar, 5

acf, 4
ar, 5
cor, 6
covOGK, 2
FastQn, 2
fqn (FastQn), 2
lmrob, 6
mad, 2, 3
nearPD, 3
posdefify, 3
psdcor, 3, 6
Qn, 3
robacf, 4
robar, 5
robcor, 6
s_FastQn, 4–6
s_FastQn (FastQn), 2
sd, 2