Package ‘CerioliOutlierDetection’

February 19, 2015

Type Package
Title Outlier detection using the iterated RMCD method of Cerioli (2010)
Version 1.0.8
Date 2014-08-01
Maintainer Christopher G. Green <christopher.g.green@gmail.com>
Depends R (>= 3.0.0)
Imports robustbase (>= 0.91-1)
Description This package provides the iterated RMCD method of Cerioli (2010) for multivariate outlier detection via robust Mahalanobis distances. It also provides the finite-sample RMCD method discussed in the paper, as well as the methods provided in Hardin and Rocke (2005) and Green and Martin (2014).
License GPL (>= 2)
Suggests rrcov, robust, mvtnorm
URL http://christopherggreen.github.io/CerioliOutlierDetection/
Author Christopher G. Green [aut, cre],
R. Doug Martin [ths]
NeedsCompilation no
Repository CRAN
Date/Publication 2014-08-02 07:29:32

R topics documented:

  cerioli2010.frmcd.test ........................................ 2
  cerioli2010.irmcd.test ........................................ 4
  CerioliOutlierDetection .................................... 7
  ch99AsymptoticDF ............................................. 8
  hr05AdjustedDF ............................................... 9
  hr05CriticalValue .......................................... 11
  hr05CutoffMvnormal ........................................ 13

Index 15
**Finite-Sample Reweighted MCD Outlier Detection Test of Cerioli (2010)**

**Description**

Given a set of observations, this function tests whether there are outliers in the data set and identifies outlying points. Outlier testing/identification is done using the Mahalanobis-distances based on the MCD dispersion estimate. The finite-sample reweighted MCD method of Cerioli (2010) is used to test for unusually large distances.

**Usage**

```r
cerioli2010.fsrmd.test(datamat, mcd.alpha = max.bdp.mcd.alpha(n,v), signif.alpha = 0.05, nsamp = 500, nmini = 300, trace = FALSE)
```

**Arguments**

- `datamat` (Data Frame or Matrix) Data set to test for outliers (rows = observations, columns = variables). `datamat` cannot have missing values; please deal with them prior to calling this function. `datamat` will be converted to a matrix.
- `mcd.alpha` (Numeric) Value to control the fraction of observations used to compute the determinant. Default value is corresponds to the maximum breakpoint case of the MCD; valid values are between 0.5 and 1. See the `covMcd` documentation in the `robustbase` library.
- `signif.alpha` (Numeric) Desired nominal size of the individual outlier test (e.g., 0.05). Equivalent, significance level at which to test observations for outlyingness. To test the intersection hypothesis of no outliers in the data, specify $\alpha = 1 - (1 - \gamma)^{1/n}$, where $\gamma$ is the nominal size of the test and $n$ is the number of observations.
- `nsamp` (Integer) Number of subsamples to use in computing the MCD. See `covMcd` documentation in the `robustbase` library.
- `nmini` (Integer) See `covMcd` documentation in the `robustbase` library.
- `trace` (Logical) See `covMcd` documentation in the `robustbase` library.

**Value**

- `mu.hat` Location estimate from the MCD calculation
- `sigma.hat` Dispersion estimate from the MCD calculation
- `mahdist` Mahalanobis distances calculated using the MCD estimate
- `DD` Hardin-Rocke critical values for testing MCD distances. Used to produce weights for reweighted MCD. See Equation (16) in Cerioli (2010).
weights
mu.hat.rw
sigma.hat.rw
mahdist.rw
critvalfcn
signif.alpha
mcd.alpha
outliers

**Author(s)**

Written and maintained by Christopher G. Green <christopher.g.green@gmail.com>

**References**


**See Also**

cerioli2010.irmcd.test

**Examples**

```r
require(mvtnorm, quiet=TRUE)

# dimension v, number of observations n
v <- 5
n <- 200
simdata <- array( rmvnorm(n*v, mean=rep(0,v),
  sigma = diag(rep(1,v))), c(n,v) )

# detect outliers with nominal sizes
# c(0.05,0.01,0.001)
# sa <- 1. - ((1. - c(0.05,0.01,0.001))^((1./n))
results <- cerioli2010.fsrmd.test( simdata,
...)
```
signif.alpha=sa )

# count number of outliers detected for each
# significance level
colSums( results$outliers )

#########################################################
# add some contamination to illustrate how to
# detect outliers using the fsrmcd test
# 10/200 = 5% contamination
simdata[ sample(n,10), ] <- array(
  rmvnorm( 10*v, mean=rep(2,v), sigma = diag(rep(1,v))),
  c(10,v)
)
results <- cerioli2010.fsrmcd.test( simdata,
  signif.alpha=sa )
colMeans( results$outliers )

## Not run:
#########################################################
# example of how to ensure the size of the test is correct

n.sim <- 5000
simdata <- array(
  rmvnorm(n*v*n.sim, mean=rep(0,v), sigma=diag(rep(1,v))),
  c(n,v,n.sim)
)
# in practice we'd do this using one of the parallel processing
# methods out there
sa <- 1. - ((1. - 0.01)^0)^1
results <- apply( simdata, 3, function(dm) {
  z <- cerioli2010.fsrmcd.test( dm,
    signif.alpha=sa )
  # true if outliers were detected in the data, false otherwise
  any(z$outliers[,1,drop=TRUE])
})
# count the percentage of samples where outliers were detected;
# should be close to the signif.alpha value used (0.01) in these
# samples
mean(results)

## End(Not run)
Description

Given a set of observations, this function tests whether there are outliers in the data set and identifies outlying points. Outlier testing/identification is done using the Mahalanobis-distances based on the MCD dispersion estimate. The iterated reweighted MCD method of Cerioli (2010) is used to ensure the test has the specified size.

Usage

cerioli2010.irmcd.test(datamat,  
mcd.alpha = max.bdp.mcd.alpha(n,v),  
signif.alpha = 0.05, nsamp = 500,  
nmini = 300, trace = FALSE)

Arguments

datamat (Data Frame or Matrix) Data set to test for outliers (rows = observations, columns = variables). datamat cannot have missing values; please deal with them prior to calling this function. datamat will be converted to a matrix.

mcd.alpha (Numeric) Value to control the fraction of observations used to compute the determinant. Default value is corresponds to the maximum breakpoint case of the MCD; valid values are between 0.5 and 1. See the covMcd documentation in the robustbase library.

signif.alpha (Numeric) Desired nominal size of the outlier test (e.g., 0.05). Equivalently, significance level at which to test observations for outlyingness. Unlike cerioli2010.fsrmcd.test, this function will calculate the appropriate significance level for the intersection hypothesis based on signif.alpha, so signif.alpha should be the desired nominal size for detecting outliers.

nsamp (Integer) Number of subsamples to use in computing the MCD. see covMcd documentation in the robustbase library.

nmini (Integer) see covMcd documentation in the robustbase library.

trace (Logical) see covMcd documentation in the robustbase library.

Details

Calls the finite-sample reweighted MCD (FSRMCD) outlier detection function cerioli2010.fsrmcd.test first to test for the existence of any outliers in the data. If the FSRMCD method rejects the null hypothesis of no outliers in the data, individual observations are then tested for outlyingness using the critical value function returned by cerioli2010.fsrmcd.test with an adjusted significance parameter.

Value

outliers A matrix of dimension nrow(datamat) by length(signif.alpha) indicating whether each row of datamat is an outlier. The i-th column corresponds to the result of testing observations for outlyingness at significance level signif.alpha[i].
mahdist.rw    a matrix of dimension nrow(datamat) by length(signif.alpha) of Mahalanobis distances computed using the finite-sample reweighted MCD methodology in Cerioli (2010). Even though the distances do not depend on signif.alpha, there is one column per entry in signif.alpha for user convenience.

Author(s)

Written and maintained by Christopher G. Green <christopher.g.green@gmail.com>

References


See Also

cretioli2010.fsrmcd.test

Examples

require(mvtnorm, quiet=TRUE)

# dimension v, number of observations n
v <- 5
n <- 200
simdata <- array(rmvnorm(n*v, mean=rep(0,v),
    sigma = diag(rep(1,v))), c(n,v))
# detect outliers
results <- cretioli2010.irmcd.test( simdata,
    signif.alpha=c(0.05,0.01,0.001) )
# count number of outliers detected for each 
# significance level
colSums( results$outliers )

# add some contamination to illustrate how to
# detect outliers using the irmcd test
# 10/200 = 5% contamination
simdata[ sample(n,10), ] <- array(
    rmvnorm( 10*v, mean=rep(2,v), sigma = diag(rep(1,v)) ),
    c(10,v) )
results <- cretioli2010.irmcd.test( simdata,
    signif.alpha=0.01 )
mean( results$outliers[,1,drop=TRUE] )
# example of how to ensure the size of the test is correct
## Not run:
```
n.sim <- 5000
simdata <- array(
  rmvnorm(n*v*n.sim, mean=rep(0,v), sigma=diag(rep(1,v))),
  c(n,v,n.sim))
```
# in practice we'd do this using one of the parallel processing
# methods out there
```
results <- apply( simdata, 3, function(dm) {
  z <- cerioli2010.irmcd.test( dm,
    signif.alpha=0.01)
  # true if outliers were detected in the data, false otherwise
  any(z$outliers[,1,drop=TRUE])
})
```
# count the percentage of samples where outliers were detected;
# should be close to the signif.alpha value used (0.01) in these
# samples
mean(results)
## End(Not run)

---

**CerioliOutlierDetection**

*CerioliOutlierDetection: package for implementing the Iterated Reweighted MCD outlier detection method of Cerioli (2010)*

---

**Description**

Implements the outlier detection methodology of Cerioli (2010) based on Mahalanobis distances and the minimum covariance determinant (MCD) estimate of dispersion. Also provides critical values for testing outlyingness of MCD-based Mahalanobis distances using the distribution approximations developed by Hardin and Rocke (2005) and Green and Martin (2014).

**Details**

<table>
<thead>
<tr>
<th>Package</th>
<th>CerioliOutlierDetection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Package</td>
</tr>
<tr>
<td>Version</td>
<td>1.0.8</td>
</tr>
</tbody>
</table>

The function `cerioli2010.irmcd.test()` provides the outlier detection methodology of Cerioli (2010), and is probably the best place for a new user of this package to start. See the documentation for that function for examples.

This package was also used to produce the results presented in Green and Martin (2014).
Author(s)

Written and maintained by Christopher G. Green <christopher.g.green@gmail.com>, with advice and support from Doug Martin.

References


Description

Computes the asymptotic Wishart degrees of freedom and consistency constant for the MCD robust dispersion estimate (for data with a model normal distribution) as described in Hardin and Rocke (2005) and using the formulas described in Croux and Haesbroeck (1999).

Usage

ch99AsymptoticDF(n.obs, p.dim, mcd.alpha)

Arguments

n.obs (Integer) Number of observations
p.dim (Integer) Dimension of the data, i.e., number of variables.
mcd.alpha (Numeric) Value that determines the fraction of the sample used to compute the MCD estimate. \(1 - mcd.alpha\) will be the fraction of observations that are omitted in computing the MCD estimate. The default value is

\[\frac{\lfloor (n.obs + p.dim + 1)/2 \rfloor}{n.obs},\]

which yields the maximum possible breakdown point for the MCD estimate.

Details

The consistency factor c.alpha is already available in the robustbase library as the function .MCDcons. (See the code for covMcd.) ch99AsymptoticDF uses the result of .MCDcons for consistency.

The computation of the Wishart degrees of freedom parameter \(m\) follows the Appendix of Hardin and Rocke (2005).
Value

c.alpha  the asymptotic consistency coefficient for the MCD estimate of the dispersion matrix
m.hat.asy the asymptotic degrees of freedom for the Wishart distribution approximation to the distribution of the MCD dispersion estimate

Author(s)

Written and maintained by Christopher G. Green <christopher.g.green@gmail.com>

References


Examples

# compare to table from p941 of Hardin and Rocke (2005)
ch99AsymptoticDF( 50, 5)
ch99AsymptoticDF( 100,10)
ch99AsymptoticDF( 500,10)
ch99AsymptoticDF(1000,20)

hr05AdjustedDF  Adjusted Degrees of Freedom for Testing Robust Mahalanobis Distances for Outlyingness

Description

Computes the adjusted F degrees of freedom for testing Mahalanobis distances calculated with the minimum covariance determinant (MCD) robust dispersion estimate (for data with a model normal distribution) as described in Hardin and Rocke (2005) or in Green and Martin (2014).

Usage

hr05AdjustedDF( n.obs, p.dim, mcd.alpha, m.asy, method = c("HR05", "GM14"))

Arguments

n.obs  (Integer) Number of observations
p.dim  (Integer) Dimension of the data, i.e., number of variables.
mcd.alpha  (Numeric) Value that determines the fraction of the sample used to compute the MCD estimate. Default value corresponds to the maximum breakdown point case of the MCD.
m.asy  (Numeric) Asymptotic Wishart degrees of freedom. The default value uses `ch99AsymptoticDF` to obtain the asymptotic value, but the user can also provide a pre-computed value.

method  Either "HR05" to use the method of Hardin and Rocke (2005), or "GM14" to use the method of Green and Martin (2014).

Details

Hardin and Rocke (2005) computed an approximate $F$ distribution for testing robust Mahalanobis distances, computed using the MCD estimate of dispersion, for outlyingness. This distribution improves upon the standard $\chi^2$ distribution for identifying outlying points in data set. The method of Hardin and Rocke was designed to work for the maximum breakdown point case of the MCD, where

$$\alpha = \left\lfloor \frac{(n.\,obs + p.\,dim + 1)/2}{n.\,obs} \right\rfloor.$$  

Green (2014) extended this result to MCD($\alpha$), where $\alpha$ controls the breakdown point of the estimator.

With argument `method = "HR05"` the function returns $m_{\text{pred}}$ as given in Equation 3.4 of Hardin and Rocke (2005). The Hardin and Rocke method is only supported for the maximum breakdown point case; an error will be generated for other values of `mcd.alpha`.

The argument `method = "GM14"` uses the extended methodology described in Green and Martin (2014) and is available for all values of `mcd.alpha`.

Value

Returns the adjusted $F$ degrees of freedom based on the asymptotic value, the dimension of the data, and the sample size.

Note

This function is typically not called directly by users; rather it is used in the construction of other functions.

Author(s)

Written and maintained by Christopher G. Green <christopher.g.green@gmail.com>

References


See Also

`ch99AsymptoticDF`
Examples

hr05tester <- function(n,p) {
  a <- floor((n+p+1)/2) / n
  hr05AdjustedDF( n, p, a, ch99AsymptoticDF(n,p,a)$m.hat.asy, method="HR05" )
}

# compare to m_pred in table on page 941 of Hardin and Rocke (2005)
hr05tester( 50, 5)
hr05tester( 100,10)
hr05tester( 500,10)
hr05tester(1000,20)

# using default arguments
hr05tester <- function(n,p) {
  hr05AdjustedDF( n, p, method="HR05" )
}

# compare to m_pred in table on page 941 of Hardin and Rocke (2005)
hr05tester( 50, 5)
hr05tester( 100,10)
hr05tester( 500,10)
hr05tester(1000,20)

# Green and Martin (2014) improved method
hr05tester <- function(n,p) {
  hr05AdjustedDF( n, p, method="GM14" )
}

# compare to m_sim in table on page 941 of Hardin and Rocke (2005)
hr05tester( 50, 5)
hr05tester( 100,10)
hr05tester( 500,10)
hr05tester(1000,20)

---

**hr05CriticalValue**

*Hardin and Rocke (2005) Critical Value for Testing MCD-based Mahalanobis Distances*

### Description

Hardin and Rocke (2005) provide an approximate $F$ distribution for testing whether Mahalanobis distances calculated using the MCD dispersion estimate are unusually large, and hence, indicative of outliers in the data.

### Usage

```
hr05CriticalValue(em, p.dim, signif.alpha)
```

### Arguments

- **em**  
  (Numeric) Degrees of freedom for Wishart distribution approximation to the MCD scatter matrix.


\begin{verbatim}
hr05CriticalValue

p.dim (Integer) Dimension of the data, i.e., number of variables.

signif.alpha (Numeric) Significance level for testing the null hypothesis

Details

Hardin and Rocke (2005) derived an $F$ distributional approximation for the Mahalanobis distances
of the observations that were excluded from the MCD calculation; see equation 3.2 on page 938 of
the paper.

It is assumed here that the MCD covariance estimate used in the Mahalanobis distance calculation
was adjusted by the consistency factor, so it is not included in the calculation here. (If one needs
the consistency factor it is returned by the function \texttt{ch99AsymptoticDF} in this package or by the
function \texttt{Nmcdcons} in the \texttt{robustbase} package.)

Value

The appropriate cutoff value (from the $F$ distributional approximation) for testing whether a Mahalanobis
distance is unusually large at the specified significance level.

Note

It can happen that one of the $F$ distribution parameters, $m - p + 1$, is non-positive, in which case
\texttt{qf} will return NaN. \texttt{hr05CriticalValue} will issue a warning in this case, and return NA.

Author(s)

Written and maintained by Christopher G. Green &lt;christopher.g.green@gmail.com&gt;

References

J. Hardin and D. M. Rocke. The distribution of robust distances. Journal of Computational and

See Also

\texttt{hr05AdjustedDF, hr05CutoffMvnormal}

Examples

\texttt{hr05CriticalValue( hr05AdjustedDF( 1000, 20 ), 20, 0.05 )}
\end{verbatim}
Corrected Critical Values for Testing MCD-based Mahalanobis Distances

Description

Provides critical values for testing for outlyingness using MCD-based Mahalanobis distances and the $F$ distributional approximation developed by Hardin and Rocke (2005) or the enhancement by Green and Martin (2014).

Usage

hr05CutoffMvnormal(n.obs, p.dim, mcd.alpha, signif.alpha, method = c("GM14", "HR05"), use.consistency.correction = FALSE)

Arguments

n.obs  (Integer) Number of observations
p.dim  (Integer) Dimension of the data, i.e., number of variables.
mcd.alpha  (Numeric) Value that determines the fraction of the sample used to compute the MCD estimate. Defaults to the value used in the maximum breakdown point case of the MCD.
signif.alpha  (Numeric) Significance level for testing the null hypothesis. Default value is 0.05.
method  Either "HR05" to use the method of Hardin and Rocke (2005), or "GM14" to use the method of Green and Martin (2014).
use.consistency.correction  (Logical) By default, the method does not multiply the cutoff values by the consistency correction factor for the MCD, under the assumption that the correction was applied during the calculation of the MCD-based Mahalanobis distances. Specify TRUE to add the correction factor if you need it for your application.

Details

hr05CutoffMvnormal is the typical way in which a user will calculate critical values for testing outlyingness via MCD-based Mahalanobis distances. The critical values come from the $F$ distributional approximation derived by Hardin and Rocke (2005). One can use either the corrected degrees of freedom parameter derived in that paper (which was only shown to work for the maximum breakdown point case of MCD), or the correction derived in Green and Martin (2014) for arbitrary values of mcd.alpha.

Value

cutoff.pred  Critical value based on the predicted Wishart degrees of freedom m.pred
cutoff.asy  Critical value based on the asymptotic Wishart degrees of freedom m.asy
c.alpha  The value of the consistency correction factor, $c_{\alpha}$
```
m.asy  Asymptotic Wishart degrees of freedom parameter
m.pred Predicted Wishart degrees of freedom (using the method specified in method)
n.obs  Number of observations
p.dim  Number of variables

Author(s)

Written and maintained by Christopher G. Green <christopher.g.green@gmail.com>

References


See Also

hr05CriticalValue, hr05AdjustedDF

Examples

# examples from page 941 of Hardin and Rocke
hr05CutoffMvnormal(n.obs=50, p.dim=5, signif.alpha=0.05)
hr05CutoffMvnormal(n.obs=100, p.dim=10, signif.alpha=0.05)
hr05CutoffMvnormal(n.obs=500, p.dim=10, signif.alpha=0.05)
hr05CutoffMvnormal(n.obs=1000, p.dim=20, signif.alpha=0.05)
```
Index

*Topic htest
  cerioli2010.fsrmd.test, 2
  cerioli2010.irmcd.test, 4

*Topic multivariate
  cerioli2010.fsrmd.test, 2
  cerioli2010.irmcd.test, 4
  ch99AsymptoticDF, 8
  hr05AdjustedDF, 9
  hr05CriticalValue, 11
  hr05CutoffMvnorma, 13

*Topic package
  CerioliOutlierDetection, 7

*Topic robust
  cerioli2010.fsrmd.test, 2
  cerioli2010.irmcd.test, 4
  ch99AsymptoticDF, 8
  hr05AdjustedDF, 9
  hr05CriticalValue, 11
  hr05CutoffMvnorma, 13
  cerioli2010.fsrmd.test, 2, 5, 6
  cerioli2010.irmcd.test, 3, 4
  CerioliOutlierDetection, 7
  CerioliOutlierDetection-package
    (CerioliOutlierDetection), 7
  ch99AsymptoticDF, 8, 10, 12
  covMcd, 2, 5

  hr05AdjustedDF, 9, 13, 14
  hr05CriticalValue, 11, 14
  hr05CutoffMvnorma, 12, 13