Package ‘psymonitor’

March 20, 2019

Type Package

Title Real Time Monitoring of Asset Markets: Bubbles and Crisis

Version 0.0.2

Description Apply the popular real-time monitoring strategy
proposed by Phillips, Shi and Yu (2015a,b;PSY) <doi:10.1111/iere.12132>, <doi:10.1111/iere.12131>, along with a new
bootstrap procedure designed to mitigate the potential impact of
heteroskedasticity and to effect family-wise size control in recursive
testing algorithms (Phillips and Shi, forthcoming).

License GPL-3

URL https://github.com/itamarcaspi/psymonitor

BugReports https://github.com/itamarcaspi/psymonitor/issues

Depends R (>= 3.2.0)

Imports doParallel, foreach, magrittr, parallel

Suggests ggplot2, knitr, lubridate, rmarkdown, spelling, testthat,

VignetteBuilder knitr

Encoding UTF-8

Language en-US

LazyData true

RoxygenNote 6.1.1

NeedsCompilation no

Author Peter C.B. Phillips [aut],
Shuping Shi [aut],
Itamar Caspi [aut, cre]

Maintainer Itamar Caspi <caspi.itamar@gmail.com>

Repository CRAN

Date/Publication 2019-03-20 07:00:03 UTC
Estimate the ADF statistic

ADF calculates the augmented Dickey-Fuller (ADF) test statistic with lag order set fixed or selected by AIC or BIC.

**Usage**

```r
ADF(y, IC = 0, adflag = 0)
```

**Arguments**

- `y`: A vector, The data.
- `IC`: An integer. 0 for fixed lag (default) order 1 for AIC and 2 for BIC (default = 0).
- `adflag`: An integer. Lag order when IC=0; maximum number of lags when IC>0 (default = 0).

**Value**

Numeric, ADF test statistic.

**References**


**Examples**

```r
y <- rnorm(100)
tstat <- ADF(y, IC = 0, adflag = 1)
```
ADFres

Estimate the ADF model under the null

Description

ADFres estimates the ADF model under the null with lag order selected by AIC or BIC

Usage

ADFres(y, IC, adflag)

Arguments

y A Vector. Data.
IC An integer, 0 for fixed lag order (default), 1 for AIC and 2 for BIC.
adflag An integer. Lag order when IC=0; maximum number of lags when IC>0 (default = 0).

Value

Numeric, ADF test statistic.

References


cvPSYmc

Simulate the finite sample critical values for the PSY test.

Description

cvPSYmc implements the real time bubble detection procedure of Phillips, Shi and Yu (2015a,b)

Usage

cvPSYmc(obs, swindow0, IC = 0, adflag = 0, nrep = 199, multiplicity = TRUE, Tb, useParallel = TRUE, nCores)
Arguments

- **obs**: A positive integer. The number of observations.
- **swindow**: A positive integer. Minimum window size (default = $T(0.01 + 1.8/\sqrt{T})$, where $T$ denotes the sample size).
- **IC**: An integer. 0 for fixed lag order (default), 1 for AIC and 2 for BIC (default = 0).
- **adflag**: An integer, lag order when IC=0; maximum number of lags when IC>0 (default = 0).
- **nrep**: A positive integer. Number of replications (default = 199).
- **multiplicity**: Logical. If multiplicity=TRUE, use family-wise size control in the recursive testing algorithms.
- **Tb**: A positive integer. The simulated sample size (swindow0+ controlling). Ignored if multiplicity=FALSE.
- **useParallel**: Logical. If useParallel=TRUE, use multi core computation.
- **nCores**: A positive integer. Optional. If useParallel=TRUE, the number of cores defaults to all but one.

Value

A matrix. BSADF bootstrap critical value sequence at the 90, 95 and 99 percent level.

References


Examples

```r
cv <- cvPSYwmc(80, IC = 0, adflag = 1, Tb = 30, nrep = 99, nCores = 1)
```

**cvPSYwmbot**

*Conduct the new composite bootstrapping for the PSY test.*

Description

`cvPSYwmbot` implements the new bootstrap procedure designed to detect bubbles and crisis periods while mitigating the potential impact of heteroskedasticity and to effect family-wise size control in recursive testing algorithms (Phillips and Shi, forthcoming).

Usage

```r
cvPSYwmbot(y, swindow0, IC = 0, adflag = 0, Tb, nboot = 199,
             useParallel = TRUE, nCores)
```
### disp

**Arguments**

- **y**
  - A vector. The data.
- **swindow0**
  - A positive integer. Minimum window size (default = \(T(0.01 + 1.8/\sqrt{T})\), where \(T\) denotes the sample size).
- **IC**
  - An integer. 0 for fixed lag order (default), 1 for AIC and 2 for BIC (default = 0).
- **adflag**
  - An integer, lag order when IC=0; maximum number of lags when IC>0 (default = 0).
- **Tb**
  - A positive integer. The simulated sample size (swindow0+ controlling).
- **nboot**
  - A positive integer. Number of bootstrap replications (default = 199).
- **useParallel**
  - Logical. If `useParallel=TRUE`, use multi core computation.
- **nCores**
  - A positive integer. Optional. If `useParallel=TRUE`, the number of cores defaults to all but one.

**Value**

A matrix. BSADF bootstrap critical value sequence at the 90, 95 and 99 percent level.

**References**


**Examples**

```r
y <- rnorm(80)
cv <- cvPSYwmboot(y, IC = 0, adflag = 1, Tb = 30, nboot = 99, nCores = 1)
```

---

**Description**

*disp* generates a data frame with bubble/crisis periods identified by the PSY procedure.

**Usage**

```r
disp(OT, obs)
```
Arguments

OT
A date vector. Bubbles/crisis periods identified by the spymonitor::locate function.

obs
A positive integer. The number of observations.

Value

A vector of strings with bubble/crisis periods.

References


Examples

data(spread)

y <- spread$value[150:200]
obs <- length(y)
swindow0 <- floor(obs*(0.01 + 1.8/sqrt(obs)))
dim <- obs - swindow0 + 1
Tb <- 24 + swindow0 - 1

# Estimate PSY statistics and CVs
bsadf <- PSY(y, swindow0)
quantilesbsadf <- cvPSYwmboot(y, swindow0, Tb=Tb, nboot = 49, nCores = 2)
quantile95 <- quantilesbsadf %*% matrix(1, nrow = 1, ncol = dim)

# locate bubble/crisis dates
ind95 <- (bsadf > t(quantile95[2, ])) * 1
monitorDates <- spread$date[swindow0:obs]
OT <- locate(ind95, monitorDates)

# Show bubble/crisis periods
disp(OT, obs)
**locate**

*Locate bubble/crisis periods*

**Description**

locate locate bubble/crisis periods with non-zero bubble indicator

**Usage**

locate(index, dates)

**Arguments**

index 
A vector. A dummy variable that equals 1 for a bubble/crisis period and 0 otherwise.

dates 
A vector. Dates of the time series.

**Value**

A vector. Dates identified as bubbles or crisis.

**References**


---

**Psy**

*Estimate PSY's BSADF sequence of test statistics*

**Description**

Psy implements the real time bubble detection procedure of Phillips, Shi and Yu (2015a,b)

**Usage**

Psy(y, swindow0, IC = 0, adflag = 0)
Arguments

- **y**: A vector. The data.
- **swindow0**: A positive integer. Minimum window size (default $= T(0.01 + 1.8/\sqrt{T})$, where $T$ denotes the sample size).
- **IC**: An integer. 0 for fixed lag order (default), 1 for AIC and 2 for BIC (default = 0).
- **adflag**: An integer, lag order when IC=0; maximum number of lags when IC>0 (default = 0).

Value

Vector, BSADF test statistic.

References


Examples

```r
y <- rnorm(8)
bsadf <- psy(y, IC = 0, adflag = 1)
```

---

**snp**

*S&P 500 price to dividend ratio*

Description

S&P 500 price to dividend ratio: January 1973– July 2017

Usage

snp

Format

A tibble

Source

Datastream International.

Examples

head(snp)
Description
Credit risk in the European sovereign sector: June 1997–June 2016. Proxied by an index constructed as a GDP weighted 10-year government bond yield of the GIIPS (Greece, Ireland, Italy, Portugal, and Spain) countries.

Usage
spread

Format
A tibble

Source
DataStream International.

Examples
head(spread)
Index

*Topic datasets
  snp, 8
  spread, 9

ADF, 2
ADRes, 3

cvPSYmc, 3
cvPSYwmboot, 4

disp, 5
locate, 7
PSY, 7

snp, 8
spread, 9