Package ‘eventstudies’

February 19, 2015

Type Package
Title Event study and extreme event analysis
Version 1.1
Date 2013-04-02
Author Ajay Shah, Vimal Balasubramaniam, Vikram Bahure
Maintainer Vikram Bahure <economics.vikram@gmail.com>
Depends R (>= 2.12.0), zoo, xts, boot
Description Implementation of short and long term event study methodology
License GPL-2
LazyLoad yes
NeedsCompilation no
Repository CRAN
Date/Publication 2013-05-08 15:39:06

R topics documented:

eventstudy-package ........................................... 2
ees .............................................................. 2
eesData ......................................................... 3
eesPlot .......................................................... 4
inference.Ecar .................................................... 5
inr ............................................................... 5
phys2eventtime ................................................... 6
remap.cumprod ................................................... 7
remap.cumsum .................................................. 7
remap.event.reindex .......................................... 8
SplitDates ....................................................... 9
StockPriceReturns ............................................. 9

Index 10

1
eventstudy-package  

Event study analysis.

Description

This package is used to undertake event study analysis using R.

Details

- Package: eventstudy
- Type: Package
- Version: 1.0
- Date: 2013-04-04
- License: GPL 2
- LazyLoad: yes

This package allows a dataset to be studied in an event-time frame and perform parametric / non-parametric analysis using several inference procedures.

Author(s)

Ajay Shah, Vimal Balasubramniam, Vikram Bahure
Maintainer: economics.vikram@gmail.com

ees  

Extreme events study for a time series.

Description

This function identifies the extreme events from a time series. It also generates summary statistics for clustered and unclustered extreme event distribution.

Usage

ees(input, prob.value)

Arguments

- input  
  'input' is the time-series on which extreme event analysis is done. This series should in returns format.

- prob.value  
  It is the tail value on the basis of which the extreme event are defined. For eg: prob.value of 5 will consider 5% tail on both sides.
Value

Output is in a list format with following items:

Summary statistics
Summary of the data set

Lower tail
Distribution of extreme events; Run length distribution; Quantile values; Yearly distribution; Extreme event data

Upper tail
Distribution of extreme events; Run length distribution; Quantile values; Yearly distribution; Extreme event data

Author(s)

Vikram Bahure

References

To convert number to words, code uses function `numbers2words` by John Fox and deprintize function by Miron Kursa

Examples

```r
library(eventstudies)
data(eesData)
input <- eesData$sp500
output <- ees(input, prob.value=5)
```

---

**eesData**

*It is the data-set used for event-study analysis.*

Description

It is a time series object with daily series for S&P 500 and Nifty (NSE index).

Usage

```r
data(eesData)
```

Format

An object with class attributes `zoo` containing resposne series for eventstudy.

Examples

```r
data(eesData)
```
eesPlot

Plotting clustered and unclustered extreme event study plot.

Description

This function generates an extreme event study plot by marking extreme events and fusing clustered events together. It plots event study plot for lower tail and upper tail events. Tail events are defined as per given probability value. The value of the event series and response series should be in returns.

Usage

eesPlot(z, response.series.name, event.series.name, titlestring, ylab, width, prob.value)

Arguments

z
Data object with both response and event series on which event study is to be performed
response.series.name
Column name of the series in 'z' on which response is to be observed
event.series.name
Column name of the series in 'z' on which event is to be observed
titlestring
Title for event study plot: Preferred to be response series name
ylab
Y-axis label
width
Width for event study plot
prob.value
It is the tail value on the basis of which the extreme event are defined. For eg: prob.value of 5 will consider 5% tail on both sides.

Value

It gives an extreme event study plot for very bad (lower tail) and very good (upper tail) events on event series.

Examples

library(eventstudies)
data(eesData)
eesPlot(z=eesData, response.series.name="nifty", event.series.name="sp500", titlestring="S&P500", ylab="(Cum.) change in nifty")
inference.Ecar  

*Bootstrap inference for the event study.*

**Description**

This function does bootstrap inference to generate distribution of average of all the cumulative returns time-series.

**Usage**

```r
inference.Ecar(z.e, to.plot=FALSE)
```

**Arguments**

- `z.e`: `z.e` is the first component of the list returned by the function `phys2eventtime`
- `to.plot`: This argument will generate an eventstudy plot of the inference estimated. If `to.plot` is equal to `TRUE` then function would generate the plot else it would not.

**Value**

A data frame with 3 columns, the lower confidence interval (CI), the mean and the upper CI which are the result of bootstrap inference.

**See Also**

`phys2eventtime`

**Examples**

```r
data(StockPriceReturns)
data(SplitDates)
es.results <- phys2eventtime(z=StockPriceReturns, events=SplitDates, width=5)
es.w <- window(es.results$z.e, start=-5, end=+5)
eventtime <- remap.cumsum(es.w, is.pc=FALSE, base=0)
inference.Ecar(z.e=eventtime, to.plot=FALSE)
```

---

**inr**  

*Exchange rate for Indian rupee: INR/USD*

**Description**

A sample of INR/USD rates from 1990 to 2011. It is stored as an xts object.

**Usage**

```r
data(inr)
```
Format

It is an xts object containing daily data from 1990 to 2011. The format is: chr ‘inr.rda’

Examples

data(inr)

phys2eventtime  Converts physical dates to event dates.

Description

Given a zoo time-series vector and an event date, it converts the physical date to the event date. The event date becomes zero and all other dates shift accordingly.

Usage

phys2eventtime(z, events, width = 10)

Arguments

z  Time series data for which event frame is to be generated.
events  It is a data frame with two columns: unit and when. unit has column name of which response is to measured on the event date, while when has the event date.
width  Width corresponds to the number of days on each side of the event date. For a given width, if there is any NA in the event window then the last observation is carried forward.

Value

Output is in a list format. A zoo object indexed with event time and a vector which describes the status of each unit in the original data.

Examples

data(StockPriceReturns)
data(SplitDates)
phys2eventtime(z=StockPriceReturns, events=SplitDates, width=5)
### remap.cumprod

*Returns geometric returns.*

**Description**

This function returns the geometric returns for a given zoo object.

**Usage**

```r
remap.cumprod(z, is.pc = TRUE, is.returns = TRUE, base = 100)
```

**Arguments**

- `z`: The zoo object returned from `phys2eventtime`.
- `is.pc`: Logical flag if input is a percentage.
- `is.returns`: Logical flag if input is returns.
- `base`: The base for the cumulative product.

**Value**

The frame returned has each column replaced by `base *` the cumulative product of the column.

**See Also**

`phys2eventtime`

**Examples**

```r
data(StockPriceReturns)
data(SplitDates)
es.results <- phys2eventtime(z=StockPriceReturns, events=SplitDates, width=5)
es.w <- window(es.results$z.e, start=-5, end=+5)
eventtime <- remap.cumprod(es.w, is.pc=FALSE, is.returns=TRUE, base=0)
```

---

### remap.cumsum

*This function gives the cumulative total of the values (in level or percentage form).*

**Description**

This function gives the cumulative total of the values.

**Usage**

```r
remap.cumsum(z, is.pc = TRUE, base = 0)
```
Arguments

\( z \) \( z \) is the zoo object returned from phys2eventtime.
\( \text{is.pc} \) \( \text{is.pc} \) is a logical flag to indicate that the values are in percentage form.
\( \text{base} \) \( \text{base} \) for the cumulative sum (default: \( \text{base} = 0 \)).

Value

The frame returned has each column replaced by the cumulative sum of the column.

See Also

eventDays, eventstudyData, phys2eventtime

Examples

```r
data(StockPriceReturns)
data(SplitDates)
es.results <- phys2eventtime(z=StockPriceReturns, events=SplitDates, width=5)
es.w <- window(es.results$z.e, start=-5, end=+5)
eventtime <- remap.cumsum(es.w, is.pc=FALSE, base=0)
```

Description

A function which consumes a zoo object where there are lots of events (as columns). The contents are all levels. For each column, the event date value is set to 100 and all other values are scaled accordingly. It’s input is a zoo object which is the first component of the list returned by phys2eventtime.

Usage

`remap.event.reindex(z)`

Arguments

\( z \) \( z \) is a zoo object containing input data from phys2eventtime.

Value

The function returns a zoo object which has been rescaled.

See Also

phys2eventtime
**SplitDates**

**Examples**

```r
data(StockPriceReturns)
data(SplitDates)
es.results <- phys2eventtime(z=StockPriceReturns, events=SplitDates,width=5)
es.w <- window(es.results$z.e, start=-5, end=+5)
eventtime <- remap.event.reindex(es.w)
```

**Description**

It is the data-set used for event-study analysis.

**Usage**

```r
data(SplitDates)
```

**Format**

An object with class attributes zoo containing resposne series for eventstudy and a data frame with stock split event dates.

**Examples**

```r
data(SplitDates)
```

---

**StockPriceReturns**

**It is the data-set used for event-study analysis.**

**Description**

It is the data-set used for event-study analysis.

**Usage**

```r
data(StockPriceReturns)
```

**Format**

An object with class attributes zoo containing resposne series for eventstudy.

**Examples**

```r
data(StockPriceReturns)
```
Index

* Topic **datasets**
  eesData, 3
  inr, 5
  SplitDates, 9
  StockPriceReturns, 9
* Topic **eventstudies**
  eventstudy-package, 2
* Topic **phys2eventtime**
  phys2eventtime, 6

ees, 2
eesData, 3
eesPlot, 4
eventstudy (eventstudy-package), 2
eventstudy-package, 2

inference.Ecar, 5
inr, 5

phys2eventtime, 6

remap.cumprod, 7
remap.cumsum, 7
remap.event.reindex, 8

SplitDates, 9
StockPriceReturns, 9