Package 'fxregime'

August 1, 2026

Version 1.0-5

Date 2026-08-01

Title Exchange Rate Regime Analysis

Description Exchange rate regression and structural change tools for estimating, testing, dating, and monitoring (de facto) exchange rate regimes.

LazyData yes

Depends R (>= 3.0.0), zoo, strucchange

Suggests lmtest, foreach, knitr, rmarkdown

Imports graphics, stats, car, sandwich

License GPL-2 | GPL-3

VignetteBuilder knitr

URL https://zeileis.codeberg.page/fxregime/

BugReports https://codeberg.org/zeileis/fxregime/issues

NeedsCompilation no

Author Achim Zeileis [aut, cre] (ORCID: <https://orcid.org/0000-0003-0918-3766>), Ajay Shah [ctb], Ila Patnaik [ctb], Anmol Sethy [ctb]

Maintainer Achim Zeileis <Achim.Zeileis@R-project.org>

Repository CRAN

Date/Publication 2026-08-01 09:10:09 UTC

Contents

confint.fxregimes 2
fxlm 4
fxmonitor 5
FXRatesCHF 7

fxregimes 8
fxreturns 10
fxtools 12
refit 13

Index **14**

`confint.fxregimes` *Confidence Intervals for Breaks between Exchange Rate Regimes*

Description

Confidence intervals for estimated changes/breaks between exchange rate regimes.

Usage

```
## S3 method for class 'fxregimes'
confint(object, parm = NULL, level = 0.95, breaks = NULL, meat. = NULL, ...)
```

Arguments

`object`	An object of class `"fxregimes"` as fitted by `fxregimes`.
`parm`	integer. Either `parm` or `breaks` may be set, see below.
`level`	numeric. The confidence level to be used.
`breaks`	integer. The number of breaks to be extracted from `object` for which confidence intervals should be computed.
`meat.`	function. A function for extracting the meat of a sandwich estimator from a `fxlm` object. By default, the inverse of `bread` is used, i.e., a correctly specified model is assumed.
`...`	currently not used.

Details

As the breakpoints are integers (observation numbers) the corresponding confidence intervals are also rounded to integers. The algorithm used is essentially the same as described for `confint.breakpointsfull`. The same distribution function is used, just the variance components are computed differently. Here, `bread` and `meat` (or some of its HC/HAC counterparts) are used. See Zeileis, Shah, Patnaik (2008) for more details.

Value

An object of class `"confint.fxregimes"`.

References

Zeileis A., Kleiber C., Krämer W., Hornik K. (2003), Testing and Dating of Structural Changes in Practice, *Computational Statistics and Data Analysis*, **44**, 109–123. doi:10.1016/S01679473(03)00030-6

Zeileis A., Shah A., Patnaik I. (2010), Testing, Monitoring, and Dating Structural Changes in Exchange Rate Regimes, *Computational Statistics and Data Analysis*, 54(6), 1696–1706. doi:10.1016/j.csda.2009.12.005.

See Also

fxregimes, refit, fxlm, confint.breakpointsfull

Examples

```
## load package and data
library("fxregime")
data("FXRatesCHF", package = "fxregime")

## compute returns for CNY (and explanatory currencies)
## for one year after abolishing fixed USD regime
cny <- fxreturns("CNY", frequency = "daily",
  start = as.Date("2005-07-25"), end = as.Date("2006-07-24"),
  other = c("USD", "JPY", "EUR", "GBP"))

## compute all segmented regression with minimal segment size of
## h = 20 and maximal number of breaks = 5.
reg <- fxregimes(CNY ~ USD + JPY + EUR + GBP,
  data = cny, h = 20, breaks = 5, ic = "BIC")
summary(reg)

## minimum BIC is attained for 2-segment (1-break) model
plot(reg)

## two regimes
## 1: tight USD peg
## 2: slightly more relaxed USD peg
round(coef(reg), digits = 3)
sqrt(coef(reg)[, "(Variance)"])

## inspect associated confidence intervals
ci <- confint(reg, level = 0.9)
ci
breakdates(ci)

## plot LM statistics along with confidence interval
fm <- fxlm(CNY ~ USD + JPY + EUR + GBP, data = cny)
scus <- gefp(fm, fit = NULL)
plot(scus, functional = supLM(0.1))
lines(ci)
```

fxlm *Exchange Rate Regression*

Description

Interface to lm for fitting exchange rate regression models (Frankel-Wei models).

Usage

```
fxlm(formula, data, ...)
```

Arguments

formula	a "formula" describing the linear model to be fit. For details see below.
data	a "zoo" time series.
...	arguments passed to lm.

Details

fxlm is a function for fitting exchange rate regression models also known as Frankel-Wei models. It is a simple convenience interface to lm: data is assumed to be a "zoo" series in which, by default, the first column is the dependent variable. If formula is omitted, the first column is regressed on the remaining columns in data. The main difference compared to plain lm models is that the error variance is reported as a full parameter (estimated by maximum likelihood) in the coef method and the estfun method (but currently not in the vcov method). Furthermore, the index (also known as the time stamps) of the underlying data set can be extracted by the time/index method.

Value

An object of class "fxlm" inheriting from "lm".

References

Shah A., Zeileis A., Patnaik I. (2005), What is the New Chinese Currency Regime?, Report 23, Department of Statistics and Mathematics, Wirtschaftsuniversitaet Wien, Research Report Series, November 2005. https://epub.wu.ac.at.

Zeileis A., Shah A., Patnaik I. (2010), Testing, Monitoring, and Dating Structural Changes in Exchange Rate Regimes, *Computational Statistics and Data Analysis*, 54(6), 1696–1706. doi:10.1016/j.csda.2009.12.005

See Also

lm, fxregimes

Examples

```
## load package and data
library("fxregime")
data("FXRatesCHF", package = "fxregime")

## compute returns for CNY (and explanatory currencies)
## for one year after abolishing fixed USD regime
cny <- fxreturns("CNY", frequency = "daily",
  start = as.Date("2005-07-25"), end = as.Date("2006-07-24"),
  other = c("USD", "JPY", "EUR", "GBP"))

## estimate full-sample exchange rate regression
fm <- fxlm(CNY ~ USD + JPY + EUR + GBP, data = cny)
coef(fm)
summary(fm)

## test parameter stability (with double max test)
scus <- gefp(fm, fit = NULL)
plot(scus, aggregate = FALSE)
## which shows a clear increase in the variance in March 2006

## alternative tests: Andrews' supLM ...
plot(scus, functional = supLM(0.1))
## ... or Nyblom-Hansen test (Cramer-von Mises type test)
plot(scus, functional = meanL2BB)
```

`fxmonitor` *Monitor Exchange Rate Regressions*

Description

Score-based monitoring of exchange rate regression models (Frankel-Wei models).

Usage

```
fxmonitor(formula, data, start, end = 3, alpha = 0.05, meat. = NULL)

## S3 method for class 'fxmonitor'
plot(x, which = NULL, aggregate = NULL,
  ylim = NULL, xlab = "Time", ylab = "Empirical fluctuation process",
  main = "Monitoring of FX model", ...)
```

Arguments

`formula`	a `"formula"` describing the linear model to be fit (as in `fxlm`.
`data`	a `"zoo"` time series (including history and monitoring time period).
`start`	starting time (typically in `"Date"` format) of the monitoring period.

end	end of the monitoring period (in scaled time, i.e., total length divided by length of history period).
alpha	significance level of the monitoring procedure.
meat.	optionally the meat of an alternative covariance matrix.
x	an object of class "fxmonitor" as fitted by fxmonitor.
which	name or number of parameter/process to plot.
aggregate	logical. Should the multivariate monitoring process be aggregated (using the absolute maximum)? Default is to aggregate for multivariate series.
ylim, xlab, ylab, main, ...	graphical parameters.

Details

fxmonitor is a function for monitoring exchange rate regression models (also known as Frankel-Wei models). It fits the model on the history period (before start) and computes the predicted scores (or estimating functions) on the monitoring period. The scaled and decorrelated process can be employed for monitoring as described by Zeileis (2005) using a double-maximum-type procedure with linear boundary. The critical values are interpolated from Table III in Zeileis et al. (2005).

Because the model just has to be fitted once (and not updated with every incoming observation), the interface of fxmonitor is much simpler than that of mefp: The data should just include all available observations (including history and monitoring period). Instead of updating the model with each incoming observation, the whole procedure has to be repeated.

The plot method visualizes the monitoring process along with its boundaries. The print method reports the breakdate (time of the first boundary crossing, if any), which can also be queried by the breakpoints and breakdates methods.

Value

An object of class "fxmonitor" which is a list including components:

process	the fitted empirical fluctuation process,
n	the number of observations in the history period,
formula	the formula used,
data	the data used,
monitor	start of the monitoring period,
end	end of monitoring period,
alpha	significance level of monitoring procedure,
critval	the critical value (for a linear boundary).

References

Zeileis A., Leisch F., Kleiber C., Hornik K. (2005), Monitoring Structural Change in Dynamic Econometric Models, *Journal of Applied Econometrics*, **20**, 99–121.

Zeileis A. (2005), A Unified Approach to Structural Change Tests Based on ML Scores, F Statistics, and OLS Residuals. *Econometric Reviews*, **24**, 445–466.

Shah A., Zeileis A., Patnaik I. (2005), What is the New Chinese Currency Regime?, Report 23, Department of Statistics and Mathematics, Wirtschaftsuniversitaet Wien, Research Report Series, November 2005. http://epub.wu.ac.at.

Zeileis A., Shah A., Patnaik I. (2010), Testing, Monitoring, and Dating Structural Changes in Exchange Rate Regimes, *Computational Statistics and Data Analysis*, 54(6), 1696–1706. doi:10.1016/j.csda.2009.12.005.

See Also

fxlm, fxregimes

Examples

```
## load package and data
library("fxregime")
data("FXRatesCHF", package = "fxregime")

## compute returns for CNY (and explanatory currencies)
## for one year after abolishing fixed USD regime
cny <- fxreturns("CNY", frequency = "daily",
  start = as.Date("2005-07-25"), end = as.Date("2006-07-24"),
  other = c("USD", "JPY", "EUR", "GBP"))

## monitor CNY regression as in Shah et al. (2005)
mon <- fxmonitor(CNY ~ USD + JPY + EUR + GBP,
  data = cny, start = as.Date("2005-11-01"))
mon

## visualization
plot(mon)
plot(mon, aggregate = FALSE)
plot(mon, which = "(Variance)")

## query breakpoint/date
breakpoints(mon)
breakdates(mon)
```

FXRatesCHF *Exchange Rates with CHF Unit Currency*

Description

Foreign exchange rates for 25 currencies with respect to Swiss franks (CHF) from 1971-01-04 to 2010-02-12.

Usage

```
data("FXRatesCHF")
```

Format

FXRatesCHF is a "zoo" series with "Date" time index containing 25 daily time series from 1971-01-04 to 2010-02-12. The columns correspond to the price currencies (in ISO 4217 format) with respect to CHF as the unit currency.

Details

Additionally, to the DEM and EUR series, it contains a column called DUR containing the EUR rates starting from 1999 and the (converted) DEM rates prior to 1999.

Source

Available online from the US Federal Reserve:

https://www.federalreserve.gov/releases/h10/Hist/

See Also

zoo

Examples

```
data("FXRatesCHF")
## compute/visualize USD/CNY rate
cny_usd <- na.trim(FXRatesCHF[,"CNY"]/FXRatesCHF[,"USD"])
plot(cny_usd)
```

---

fxregimes *Dating Breaks between Exchange Rate Regimes*

Description

Estimate changes/breaks between exchange rate regimes (based on exchange rate regression models).

Usage

```
fxregimes(formula, data, ..., hpc = c("none", "foreach"))
```

Arguments

formula	a "formula" describing the linear model to be fit (as in fxlm.
data	a "zoo" time series.
...	further arguments passed to gbreakpoints (currently unexported, see below), most of the arguments are identical to the formula method of breakpoints
hpc	a character specifying the high performance computing support. Default is "none", can be set to "foreach".

Details

fxregimes segments an exchange rate regression (Frankel-Wei regression) into different regimes. The breakpoints are chosen to maximize the likelihood of a Gaussian regression (with regime-specific variances). The number of breakpoints are selected according to information criteria (by default LWZ, but BIC is also reported).

The computing engine behind fxregime is gbreakpoints that generalizes various aspects about breakpoints. Its interface is not yet fixed, hence it is not exported in the namespace (and not documented). Most arguments are similar to those of the formula method of breakpoints, in particular the minimal segment size h and the maximal number of breaks breaks. To select the default information criterion the new argument ic can be set either to "LWZ" (the default) or "BIC".

A set of methods for useful generic functions is available, including plot, print, summary, lines, coef, fitted, residuals, breakfactor, breakpoints, breakdates, etc. Mostly, they behave like their "breakpoints" counterparts. Querying information about the models on each segment is made particularly easy by providing a refit method that returns a list of fxlm objects.

Optional support for high performance computing is available, currently using foreach for the dynamic programming algorithm. If hpc = "foreach" is to be used, a parallel backend should be registered before. See foreach for more information.

Value

An object of class "fxregimes" inheriting from "gbreakpointsfull", "gbreakpoints", "breakpointsfull", "breakpoints".

References

Zeileis A., Kleiber C., Krämer W., Hornik K. (2003), Testing and Dating of Structural Changes in Practice, *Computational Statistics and Data Analysis*, **44**, 109-123. doi:10.1016/S01679473(03)00030-6

Shah A., Zeileis A., Patnaik I. (2005), What is the New Chinese Currency Regime?, Report 23, Department of Statistics and Mathematics, Wirtschaftsuniversitaet Wien, Research Report Series, November 2005. https://epub.wu.ac.at.

Zeileis A., Shah A., Patnaik I. (2010), Testing, Monitoring, and Dating Structural Changes in Exchange Rate Regimes, *Computational Statistics and Data Analysis*, 54(6), 1696–1706. doi:10.1016/j.csda.2009.12.005.

See Also

fxlm, fxregimes, refit

Examples

```
## load package and data
library("fxregime")
data("FXRatesCHF", package = "fxregime")

## compute returns for CNY (and explanatory currencies)
## for one year after abolishing fixed USD regime
cny <- fxreturns("CNY", frequency = "daily",
  start = as.Date("2005-07-25"), end = as.Date("2006-07-24"),
  other = c("USD", "JPY", "EUR", "GBP"))

## compute all segmented regression with minimal segment size of
## h = 20 and maximal number of breaks = 5.
reg <- fxregimes(CNY ~ USD + JPY + EUR + GBP,
  data = cny, h = 20, breaks = 5, ic = "BIC")
summary(reg)

## minimum BIC is attained for 2-segment (1-break) model
plot(reg)

## two regimes
## 1: tight USD peg
## 2: slightly more relaxed USD peg
round(coef(reg), digits = 3)
sqrt(coef(reg)[, "(Variance)"])

## inspect two individual models by re-fitting
refit(reg)
```

`fxreturns` *Compute Exchange Rate Returns*

Description

Compute a multivariate series of exchange rate returns (log-returns in percent) from a multivariate series of exchange rates.

Usage

```
fxreturns(x, other = c("USD", "JPY", "DUR", "GBP"), data = NULL,
  frequency = "weekly", start = NULL, end = NULL, na.action = na.locf, trim = FALSE)
```

Arguments

`x`	character with column name of selected target currency.
`other`	character vector with column names of further currencies (to be used as regressors).
`data`	a `"zoo"` time series with FX rates (with respect to a base currency). By default `FXRatesCHF` is used.

frequency	character specifying whether weekly or daily returns should be computed.
start	start time of the exchange rate series (before computing returns).
end	end time of the exchange rate series.
na.action	function for handling NAs.
trim	logical or numeric. Should time points with extreme returns of the target currency be excluded? If set to TRUE, the quantiles c(0.01, 0.99) are used for trimming.

Details

fxreturns is a convenience function for selecting a smaller number of currencies from a large database of exchange rates (such as FXRatesCHF provided with the package), selecting a certain time window (if necessary), computing returns (weekly or daily), eliminating missing values and potentially trimming extreme returns (of the target currency).

Value

A "zoo" with the returns of the specified currencies.

References

Shah A., Zeileis A., Patnaik I. (2005), What is the New Chinese Currency Regime?, Report 23, Department of Statistics and Mathematics, Wirtschaftsuniversitaet Wien, Research Report Series, November 2005. https://epub.wu.ac.at.

Zeileis A., Shah A., Patnaik I. (2010), Testing, Monitoring, and Dating Structural Changes in Exchange Rate Regimes, *Computational Statistics and Data Analysis*, 54(6), 1696–1706. doi:10.1016/j.csda.2009.12.005.

See Also

fxlm, fxregimes, fxmonitor

Examples

```
## load package and data
library("fxregime")
data("FXRatesCHF", package = "fxregime")

## compute returns for CNY (and explanatory currencies)
## for one year after abolishing fixed USD regime
cny <- fxreturns("CNY", frequency = "daily",
  start = as.Date("2005-07-25"), end = as.Date("2006-07-24"),
  other = c("USD", "JPY", "EUR", "GBP"))
plot(cny)
```

fxtools *Various Tools for Exchange Rate Regime Classification*

Description

Tools for exchange rate regime classification, currently under development.

Usage

```
fxpegtest(model, peg = NULL, ...)
```

Arguments

model	an object of class "fxlm" as returned by fxlm.
peg	character with the name of the currency the target currency is pegged to. By default this is chosen to be the currency with the maximal absolute coefficient.
...	arguments passed to linearHypothesis.

Details

These tools should help to automate exchange rate regime classification.

The first building block is the function fxpegtest, a simple convenience interface to linearHypothesis. It assess the null hypothesis that only the peg currency has coefficient 1 and all other currencies have coefficient 0.

References

Shah A., Zeileis A., Patnaik I. (2005), What is the New Chinese Currency Regime?, Report 23, Department of Statistics and Mathematics, Wirtschaftsuniversitaet Wien, Research Report Series, November 2005. https://epub.wu.ac.at.

Zeileis A., Shah A., Patnaik I. (2010), Testing, Monitoring, and Dating Structural Changes in Exchange Rate Regimes, *Computational Statistics and Data Analysis*, 54(6), 1696–1706. doi:10.1016/j.csda.2009.12.005.

See Also

fxlm, fxregimes

Examples

```
## load package and data
library("fxregime")
data("FXRatesCHF", package = "fxregime")

## compute returns for CNY (and explanatory currencies)
## after abolishing fixed USD regime until end of 2005
cny <- fxreturns("CNY", frequency = "daily",
```

```
  start = as.Date("2005-07-25"), end = as.Date("2005-12-31"),
  other = c("USD", "JPY", "EUR", "GBP"))

## estimate full-sample exchange rate regression model
fm <- fxlm(CNY ~ USD + JPY + EUR + GBP, data = cny)

## check for plain USD peg:
fxpegtest(fm)
## no deviation from a plain USD peg
```

`refit` *Refitting Models*

Description

Generic function for refitting a models on various subsets or reweighted data sets.

Usage

```
refit(object, ...)
```

Arguments

`object`	an object.
`...`	arguments passed to methods.

Details

`refit` is a new generic function for refitting a certain model object on multiple versions of a data set (and is hence different from `update`). Applications are partitioned/segmented models where the same type of model should be refitted on each sub-sample or partition. See `fxregimes` for such an example. Another application would be reweighted models such as in mixture models.

The generic is similar to the one provided in **modeltools** (and should fulfill the same purpose). To avoid dependence on this package, it is also provided here.

See Also

`fxregimes`

Index

∗ **datasets**
FXRatesCHF, 7
∗ **regression**
confint.fxregimes, 2
fxlm, 4
fxmonitor, 5
fxregimes, 8
fxreturns, 10
fxtools, 12
refit, 13

AIC.gbreakpoints *(*fxregimes*)*, 8
AIC.gbreakpointsfull *(*fxregimes*)*, 8

bread, *2*
bread.fxlm *(*fxlm*)*, 4
breakdates.fxmonitor *(*fxmonitor*)*, 5
breakdates.gbreakpoints *(*fxregimes*)*, 8
breakddates.confint.fxregimes
*(*confint.fxregimes*)*, 2
breakpoints, *9*
breakpoints.fxmonitor *(*fxmonitor*)*, 5
breakpoints.gbreakpointsfull
*(*fxregimes*)*, 8

coef.fxlm *(*fxlm*)*, 4
coef.fxregimes *(*fxregimes*)*, 8
confint.breakpointsfull, *2, 3*
confint.fxregimes, 2
confint.gbreakpoints *(*fxregimes*)*, 8

estfun.fxlm *(*fxlm*)*, 4

fitted.fxregimes *(*fxregimes*)*, 8
foreach, *9*
fxlm, *2, 3*, 4, *5*, *7*, *9*, *11, 12*
fxmonitor, 5, *11*
fxpegtest *(*fxtools*)*, 12
FXRatesCHF, 7, *10, 11*
fxregimes, *2–4*, *7*, 8, *9*, *11–13*
fxreturns, 10
fxtools, 12

gbreakpoints *(*fxregimes*)*, 8

index.fxlm *(*fxlm*)*, 4
index.fxregimes *(*fxregimes*)*, 8

linearHypothesis, *12*
lines.confint.fxregimes
*(*confint.fxregimes*)*, 2
lines.fxregimes *(*fxregimes*)*, 8
lm, *4*
logLik.gbreakpoints *(*fxregimes*)*, 8

meat, *2*
mefp, *6*

plot.fxmonitor *(*fxmonitor*)*, 5
plot.gbreakpointsfull *(*fxregimes*)*, 8
plot.summary.gbreakpointsfull
*(*fxregimes*)*, 8
print.confint.fxregimes
*(*confint.fxregimes*)*, 2
print.fxmonitor *(*fxmonitor*)*, 5
print.fxregimes *(*fxregimes*)*, 8
print.gbreakpoints *(*fxregimes*)*, 8
print.summary.gbreakpointsfull
*(*fxregimes*)*, 8

refit, *3*, *9*, 13
refit.fxregimes *(*fxregimes*)*, 8
residuals.fxregimes *(*fxregimes*)*, 8

summary.gbreakpoints *(*fxregimes*)*, 8
summary.gbreakpointsfull *(*fxregimes*)*, 8

time.fxlm *(*fxlm*)*, 4
time.fxregimes *(*fxregimes*)*, 8

vcov, *4*

zoo, *8*