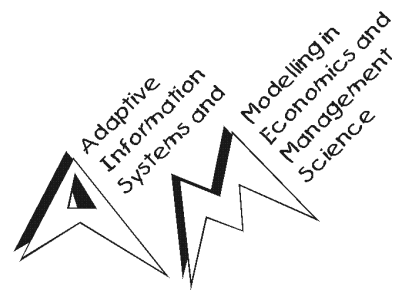


Report Series



strucchange: An R Package for Testing for Structural Change in Linear Regression Models

Achim Zeileis, Friedrich Leisch, Kurt Hornik,
Christian Kleiber

Report No. 55
May 2001

Report Series



May 2001

SFB
'Adaptive Information Systems and Modelling in Economics and Management
Science'

Vienna University of Economics
and Business Administration
Augasse 2–6, 1090 Wien, Austria

in cooperation with
University of Vienna
Vienna University of Technology

<http://www.wu-wien.ac.at/am>

Papers published in this report series
are preliminary versions of journal articles.

This paper was accepted for publication in:
Journal of Statistical Software, 7(2), 1–38,
<http://www.jstatsoft.org/v07/i02/>

This piece of research was supported by the Austrian Science Foundation
(FWF) under grant SFB#010 ('Adaptive Information Systems and Modelling in
Economics and Management Science').

strucchange: An R Package for Testing for Structural Change in Linear Regression Models

Achim Zeileis Friedrich Leisch Kurt Hornik
Institut für Statistik & Wahrscheinlichkeitstheorie
Technische Universität Wien

Christian Kleiber
Institut für Wirtschafts- und Sozialstatistik
Universität Dortmund

Abstract

This paper introduces ideas and methods for testing for structural change in linear regression models and presents how these have been realized in an R package called `strucchange`¹. It features tests from the generalized fluctuation test framework as well as from the F test (Chow test) framework. Extending standard significance tests it contains methods to fit, plot and test empirical fluctuation processes (like CUSUM, MOSUM and estimates-based processes) on the one hand and to compute, plot and test sequences of F statistics with the $\text{sup}F$, $\text{ave}F$ and $\text{exp}F$ test on the other. Thus, it makes powerful tools available to display information about structural changes in regression relationships and to assess their significance. Furthermore it is described how incoming data can be monitored online.

Keywords: structural change, CUSUM, MOSUM, recursive estimates, moving estimates, online monitoring, R, S.

1 Introduction

The problem of detecting structural changes in linear regression relationships has been an important topic in statistical and econometric research. The most important classes of tests on structural change are the tests from the generalized fluctuation test framework (Kuan and Hornik, 1995) on the one hand and tests based on F statistics (Hansen, 1992; Andrews, 1993; Andrews and Ploberger, 1994) on the other. The first class includes in particular the CUSUM and MOSUM tests and the fluctuation test, while the Chow and the $\text{sup}F$ test belong to the latter. A topic that gained more interest rather recently is to monitor structural change, i.e., to start after a history phase (without structural changes) to analyze new observations and to be able to detect a structural change as soon after its occurrence as possible.

This paper concerns ideas and methods for implementing generalized fluctuation tests as well as F tests in a comprehensive and flexible way, that reflects the common features of the testing procedures and offers facilities to display the results in various ways. These ideas have been realized in a package called `strucchange` in the R system² for statistical computing, the GNU implementation of the S language.

This paper is organized as follows: In Section 2 the standard linear regression model upon which all tests are based will be described and the testing problem will be specified. Section 3 introduces a data set which is also available in the package and which is used for the examples in this paper. The following sections

¹The package is available in the Comprehensive R Archive Network at <http://cran.r-project.org>.

²<http://www.r-project.org/>

4, 6 and 5 will then explain the tests, how they are implemented in `strucchange` and give examples for each. Section 4 is concerned with computing empirical fluctuation processes, plotting them and the corresponding boundaries and finally testing for structural change based on these processes. Analogously Section 5 introduces the F statistics and their plotting and testing methods before Section 6 extends the tools from Section 4 for the monitoring case.

2 The model

Consider the standard linear regression model

$$y_i = x_i^\top \beta_i + u_i \quad (i = 1, \dots, n), \quad (1)$$

where at time i , y_i is the observation of the dependent variable, $x_i = (1, x_{i2}, \dots, x_{ik})^\top$ is a $k \times 1$ vector of observations of the independent variables, with the first component equal to unity, u_i are iid($0, \sigma^2$), and β_i is the $k \times 1$ vector of regression coefficients. Tests on structural change are concerned with testing the null hypothesis of “no structural change”

$$H_0 : \quad \beta_i = \beta_0 \quad (i = 1, \dots, n) \quad (2)$$

against the alternative that the coefficient vector varies over time, with certain tests being more or less suitable (i.e., having good or poor power) for certain patterns of deviation from the null hypothesis.

It is assumed that the regressors are nonstochastic with $\|x_i\| = O(1)$ and that

$$\frac{1}{n} \sum_{i=1}^n x_i x_i^\top \longrightarrow Q \quad (3)$$

for some finite regular matrix Q . These are strict regularity conditions excluding trends in the data which are assumed for simplicity. For some tests these assumptions can be extended to dynamic models without changing the main properties of the tests. But as these details are not part of the focus of this work they are omitted here.

Because most of the tests on structural change considered here are either based on estimates of the regression coefficients or on residuals, the most important ones are presented here for ease of notation. $\hat{\beta}^{(i,j)}$ is the ordinary least squares (OLS) estimate of the regression coefficients based on the observations $i + 1, \dots, i + j$, and $\hat{\beta}^{(i)} = \hat{\beta}^{(0,i)}$ is the OLS estimate based on all observations up to i . Hence $\hat{\beta}^{(n)}$ is the common OLS estimate in the linear regression model. Similarly $X^{(i)}$ is the regressor matrix based on all observations up to i . The OLS residuals are denoted $\hat{u}_i = y_i - x_i^\top \hat{\beta}^{(n)}$ with the variance estimate $\hat{\sigma}^2 = \frac{1}{n-k} \sum_{i=1}^n \hat{u}_i^2$. Another type of residuals that are often used in tests on structural change are the recursive residuals

$$\tilde{u}_i = \frac{y_i - x_i^\top \hat{\beta}^{(i-1)}}{\sqrt{1 + x_i^\top (X^{(i-1)\top} X^{(i-1)})^{-1} x_i}} \quad (i = k + 1, \dots, n), \quad (4)$$

which have zero mean and variance σ^2 under the null hypothesis. The corresponding variance estimate is $\tilde{\sigma}^2 = \frac{1}{n-k} \sum_{i=k+1}^n (\tilde{u}_i - \bar{\tilde{u}})^2$.

3 The data

The data used for examples throughout this paper are macroeconomic time series from the USA. The data set contains the monthly personal income and personal consumption expenditures (in billion US dollars) between January 1959 and February 2001, which are seasonally adjusted at annual rates. It is available in the `strucchange` package: the data can be loaded and a suitable subset chosen by

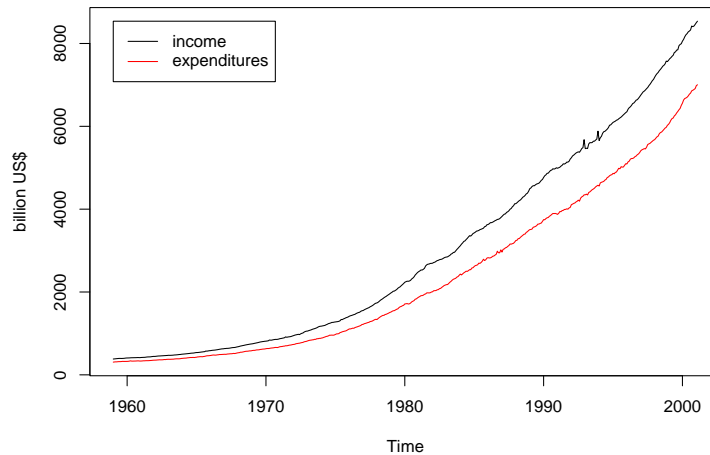


Figure 1: Personal income and personal consumption expenditures in the US

```
R> library(strucchange)
R> data(USIncExp)
R> USIncExp2 <- window(USIncExp, start=c(1970,1), end=c(1989,12))
```

It was originally taken from <http://www.economagic.com/>, a web site for economic times series. Both time series can be seen in Figure 1.

We choose our regression model to have one “real” regressor `income` (and a constant) and a dependent variable `expenditure` and we want to investigate whether the way the income is spent changes over time. From a macroeconomic view it would probably be more correct to assume that the income is influenced by the expenditure, but our approach might be more intuitive and the results can be interpreted in the same way. We will investigate whether the way the income was spent for personal consumption changed during the 1970s and 1980s. To get a first impression of the data the ratio of expenditure and income is shown for this period in Figure 2. From this it can be seen that most of the time around 77 to 78 percent of the income were spent on consumption (ignoring the constant), but that there are two peaks: a smaller one around 1974 and a larger one around 1982. We suspect that these might be structural changes and we will come back to this question in the examples in the following sections.

4 Generalized fluctuation tests

The generalized fluctuation tests fit a model to the given data and derive an empirical process, that captures the fluctuation either in residuals or in estimates. Under the null hypothesis the fluctuation should not be too large, such that for each process type boundaries can be computed, whose crossing probability under the null hypothesis is α . If the empirical process path crosses these boundaries, the fluctuation is improbably large and hence the hypothesis should be rejected (at confidence level α).

4.1 Empirical fluctuation processes: function `eFP`

Given a formula that describes a linear regression model to be tested the function `eFP` creates an object of class “`eFP`” which contains a fitted empirical fluctuation process of a specified type. The types available

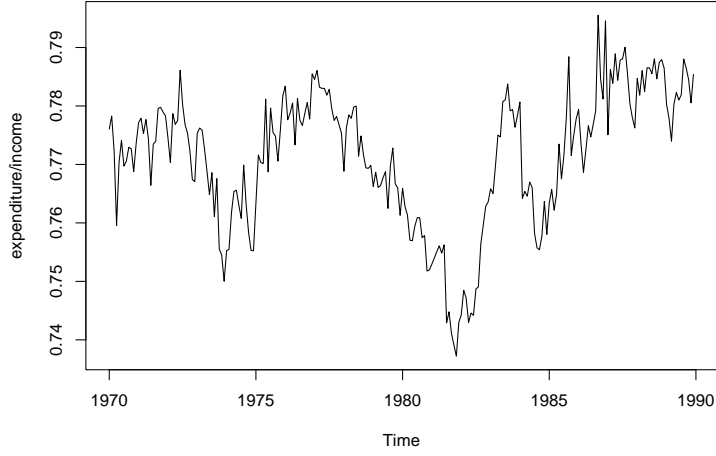


Figure 2: Relative Expenditure

will be described in detail in this section.

CUSUM processes: The first type of processes that can be computed are CUSUM processes, which contain cumulative sums of standardized residuals. [Brown et al. \(1975\)](#) suggested to consider cumulative sums of recursive residuals:

$$W_n(t) = \frac{1}{\hat{\sigma}\sqrt{\eta}} \sum_{i=k+1}^{k+\lfloor t\eta \rfloor} \tilde{u}_i \quad (0 \leq t \leq 1), \quad (5)$$

where $\eta = n - k$ is the number of recursive residuals and $\lfloor t\eta \rfloor$ is the integer part of $t\eta$.

Under the null hypothesis the limiting process for the empirical fluctuation process $W_n(t)$ is the Standard Brownian Motion (or Wiener Process) $W(t)$. More precisely the following functional central limit theorem (FCLT) holds:

$$W_n \Rightarrow W, \quad (6)$$

as $n \rightarrow \infty$, where \Rightarrow denotes weak convergence of the associated probability measures.

Under the alternative, if there is just a single structural change point t_0 , the recursive residuals will only have zero mean up to t_0 . Hence the path of the process should be close to 0 up to t_0 and leave its mean afterwards. [Krämer et al. \(1988\)](#) show that the main properties of the CUSUM quantity remain even under weaker assumptions, in particular in dynamic models. Therefore `eFp` has the logical argument `dynamic`; if set to `TRUE` the lagged observations x_{t-1} will be included as regressors.

[Ploberger and Krämer \(1992\)](#) suggested to base a structural change test on cumulative sums of the common OLS residuals. Thus, the OLS-CUSUM type empirical fluctuation process is defined by:

$$W_n^0(t) = \frac{1}{\hat{\sigma}\sqrt{n}} \sum_{i=1}^{\lfloor nt \rfloor} \hat{u}_i \quad (0 \leq t \leq 1). \quad (7)$$

The limiting process for $W_n^0(t)$ is the standard Brownian bridge $W^0(t) = W(t) - tW(1)$. It starts in 0 at $t = 0$ and it also returns to 0 for $t = 1$. Under a single structural shift alternative the path should have a

peak around t_0 .

These processes are available in the function `eFP` by specifying the argument `type` to be either "Rec-CUSUM" or "OLS-CUSUM", respectively.

MOSUM processes: Another possibility instead of using cumulative sums of residuals to detect a structural change is to analyze moving sums of the same residuals. The resulting empirical fluctuation process does then not contain the sum of all residuals up to a certain time t but the sum of a fixed number of residuals in a data window whose size is determined by the bandwidth parameter $h \in (0, 1)$ and which is moved over the whole sample period. Hence the Recursive MOSUM process is defined by

$$M_n(t|h) = \frac{1}{\hat{\sigma}\sqrt{\eta}} \sum_{i=k+\lfloor N_\eta t \rfloor+1}^{k+\lfloor N_\eta t \rfloor+\lfloor \eta h \rfloor} \tilde{u}_i \quad (0 \leq t \leq 1-h) \quad (8)$$

$$= W_n \left(\frac{\lfloor N_\eta t \rfloor + \lfloor \eta h \rfloor}{\eta} \right) - W_n \left(\frac{\lfloor N_\eta t \rfloor}{\eta} \right), \quad (9)$$

where $N_\eta = (\eta - \lfloor \eta h \rfloor)/(1-h)$. Similarly the OLS-based MOSUM process is defined by

$$M_n^0(t|h) = \frac{1}{\hat{\sigma}\sqrt{n}} \left(\sum_{i=\lfloor N_n t \rfloor+1}^{\lfloor N_n t \rfloor+\lfloor nh \rfloor} \hat{u}_i \right) \quad (0 \leq t \leq 1-h) \quad (10)$$

$$= W_n^0 \left(\frac{\lfloor N_n t \rfloor + \lfloor nh \rfloor}{n} \right) - W_n^0 \left(\frac{\lfloor N_n t \rfloor}{n} \right), \quad (11)$$

where $N_n = (n - \lfloor nh \rfloor)/(1-h)$. As the representations (9) and (11) suggest, the limiting process for the empirical MOSUM processes are the increments of a Brownian motion or a Brownian bridge respectively. This is shown in detail in [Chu et al. \(1995a\)](#).

If again a single structural shift is assumed at t_0 , then both MOSUM paths should also have a strong shift around t_0 .

The MOSUM processes will be computed if `type` is set to "Rec-MOSUM" or "OLS-MOSUM", respectively.

Estimates-based processes: Alternatively to defining fluctuation processes on the basis of residuals they can be equally well based on estimates of the unknown regression coefficients. With the same ideas as for the residual-based CUSUM- and MOSUM-type processes the $k \times 1$ -vector β is either estimated recursively with a growing number of observations or with a moving data-window of constant bandwidth h and then compared to the estimates based on the whole sample. The first idea leads to the fluctuation process which is defined by

$$Y_n(t) = \frac{\sqrt{i}}{\hat{\sigma}\sqrt{n}} \left(X^{(i)\top} X^{(i)} \right)^{\frac{1}{2}} \left(\hat{\beta}^{(i)} - \hat{\beta}^{(n)} \right), \quad (12)$$

where $i = \lfloor k + t(n-k) \rfloor$ with $t \in [0, 1]$. And the latter gives the moving estimates (ME) process

$$Z_n(t|h) = \frac{\sqrt{\lfloor nh \rfloor}}{\hat{\sigma}\sqrt{n}} \left(X^{\lfloor nt \rfloor, \lfloor nh \rfloor \top} X^{\lfloor nt \rfloor, \lfloor nh \rfloor} \right)^{\frac{1}{2}} \left(\hat{\beta}^{\lfloor nt \rfloor, \lfloor nh \rfloor} - \hat{\beta}^{(n)} \right), \quad (13)$$

where $0 \leq t \leq 1-h$. Both are k -dimensional empirical processes. Thus the limiting processes are a k -dimensional Brownian Bridge or the increments thereof respectively. Instead of rescaling the processes for each i they can also be standardized by $\left(X^{(n)\top} X^{(n)} \right)^{\frac{1}{2}}$. This has the advantage that it has to be calculated only once, but [Kuan and Chen \(1994\)](#) showed that if there are dependencies between the regressors the

rescaling improves the empirical size of the resulting test. Heuristically the rescaled empirical fluctuation process looks more like its theoretic counterpart.

Under a single shift alternative the recursive estimates processes should have a peak and the moving estimates process should again have a shift close to the shift point t_0 .

For `type="fluctuation"` the function `efp` returns the recursive estimates process, whereas set to `"ME"` the moving estimates process is returned.

These six processes can be fitted easily using the function `efp`. For our example we want to fit an OLS-based CUSUM process, and a moving estimates (ME) process with bandwidth $h = 0.2$. The commands are simply

```
R> ocus <- efp(expenditure~income, type="OLS-CUSUM", data=USIncExp2)
R> me <- efp(expenditure~income, type="ME", data=USIncExp2, h=0.2)
```

This would return objects of class `"efp"` which contain mainly the empirical fluctuation processes and a few further attributes like the process type. The process itself is of class `"ts"` (the time series class in R), which either preserves the time properties of the dependent variable if this is a time series (like in our example), or which is standardized to the interval $[0, 1]$ (or a subinterval). For the MOSUM- and ME-processes the centered interval $[h/2, 1 - h/2]$ is chosen rather than $[0, 1 - h]$.

Any other process type introduced in this section can be fitted by setting the `type` argument. The fitted process can then be printed, plotted or tested with the corresponding test on structural change. For the latter appropriate boundaries are needed; the concept of boundaries for fluctuation processes is explained in the next section.

4.2 Boundaries and plotting

The idea that is common to all generalized fluctuation tests is that the null hypothesis of “no structural change” should be rejected when the fluctuation of the empirical process $efp(t)$ gets improbably large compared to the fluctuation of the limiting process. For the one-dimensional residual-based processes this comparison is performed by some appropriate boundary $b(t)$, that the limiting process just crosses with a given probability α . Thus, if $efp(t)$ crosses either $b(t)$ or $-b(t)$ for any t then it has to be concluded that the fluctuation is improbably large and the null hypothesis can be rejected at confidence level α . The procedure for the k -dimensional estimates-based processes is similar, but instead of a boundary for the process itself a boundary for $\|efp(t)\|$ is used, where $\|\cdot\|$ is an appropriate functional. We have implemented the functionals ‘max’ and ‘range’. The null hypothesis is rejected if $\|efp(t)\|_i$ gets larger than a constant λ , which depends on the confidence level α , for any $i = 1, \dots, k$.

The boundaries for the MOSUM processes are also constants, i.e., of form $b(t) = \lambda$, which seems natural as the limiting processes are stationary. The situation for the CUSUM processes is different though. Both limiting processes, the Brownian motion and the Brownian bridge, respectively, are not stationary. It would seem natural to use boundaries that are proportional to the standard deviation function of the corresponding theoretic process, i.e.,

$$b(t) = \lambda \cdot \sqrt{t} \tag{14}$$

$$b(t) = \lambda \cdot \sqrt{t(1-t)} \tag{15}$$

for the Recursive CUSUM and the OLS-based CUSUM path respectively, where λ determines the confidence level. But the boundaries that are commonly used are linear, because a closed form solution for the crossing probability is known. So the standard boundaries for the two process are of type

$$b(t) = \lambda \cdot (1 + 2t) \tag{16}$$

$$b(t) = \lambda. \tag{17}$$

They were chosen because they are tangential to the boundaries (14) and (15) respectively in $t = 0.5$. However, Zeileis (2000a) examined the properties of the alternative boundaries (14) and (15) and showed that the power of the OLS-based CUSUM test can be improved for structural changes early and late in the sample period.

Given a fitted empirical fluctuation process the boundaries can be computed very easily using the function `boundary`, which returns a time series object with the same time properties as the given fluctuation process:

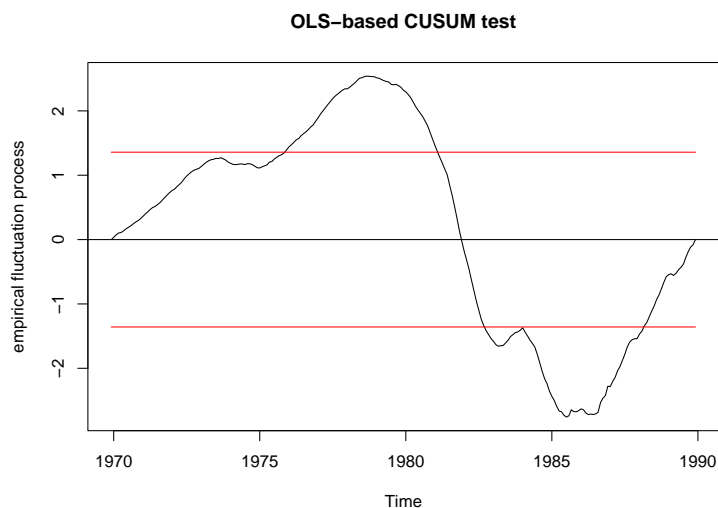


Figure 3: OLS-based CUSUM process

```
R> bound.ocus <- boundary(ocus, alpha=0.05)
```

It is also rather convenient to plot the process with its boundaries for some confidence level α (by default 0.05) to see whether the path exceeds the boundaries or not: The result of

```
R> plot(ocus)
```

is shown in Figure 3. It can be seen that the OLS-based CUSUM path has two peaks which correspond to the two peaks we have seen in Figure 2. So the two structural changes are detected, but with a short lag of a few observations.

It is also possible to suppress the boundaries and add them afterwards, e.g. in another colour

```
R> plot(ocus, boundary=FALSE)
R> lines(bound.ocus, col=4)
R> lines(-bound.ocus, col=4)
```

For residual-based processes it is only sensible to do time series plots if the functional ‘max’ is used because it is equivalent to reject the null hypothesis when $\max_{i=1,\dots,k} \|efp_i(t)\|$ gets large or when the $\max_t \max_{i=1,\dots,k} efp_i(t)$ gets large. This again is equivalent to any one of the (one-dimensional) processes $efp_i(t)$ for $i = 1, \dots, k$ exceeding the boundary. The k -dimensional process can also be plotted by specifying the parameter functional (which defaults to "max") as NULL:

```
R> plot(me, functional=NULL)
```

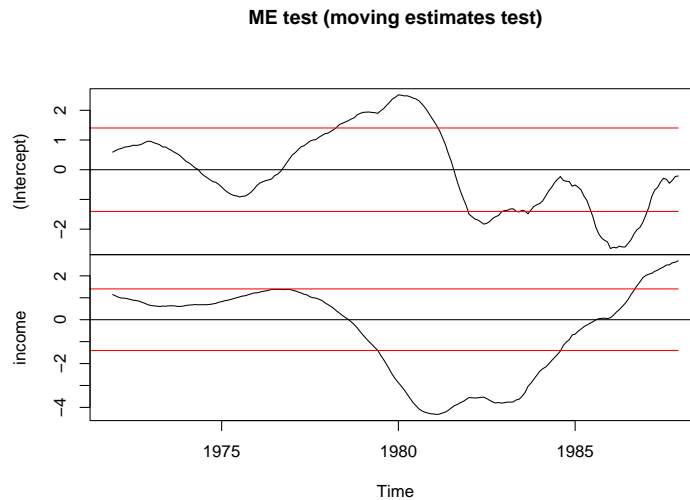


Figure 4: 2-dimensional moving estimates process

The output from R can be seen in Figure 4, where the upper and lower half show the processes that correspond to the estimate of the intercept and the income regression coefficient, respectively. Both paths have two strong shifts that cross the upper and the lower boundary. The first shift starts in about 1975/6 and the second around 1981/2. Thus, the ME test leads to similar results as the OLS-based CUSUM test. Both indicate two strong shifts, one in the mid-70s and another one in the early 80s.

4.3 Significance testing with empirical fluctuation processes

Although calculating and plotting the empirical fluctuation process with its boundaries provides and visualizes most of the information, it might still be necessary or desirable to carry out a traditional significance test. This can be done easily with the function `sctest` (structural change test) which returns an object of class "htest" containing in particular the test statistic and the corresponding p value. The test statistics reflect what was described by the crossing of boundaries in the previous section. Hence the test statistic is S_r from (18) for the residual-based processes and S_e from (19) for the estimates-based processes:

$$S_r = \max_t \frac{efp(t)}{f(t)}, \quad (18)$$

$$S_e = \max ||efp(t)||, \quad (19)$$

where $f(t)$ depends on the shape of the boundary, i.e., $b(t) = \lambda \cdot f(t)$. For most boundaries is $f(t) \equiv 1$ but the linear boundary for the Recursive CUSUM test has shape $f(t) = 1 + 2t$, e.g.

It is either possible to supply `sctest` with a fitted empirical fluctuation process or with a formula describing the model that should be tested. Thus, the commands

```
R> sctest(ocus)
R> sctest(expenditure~income, type="OLS-CUSUM", data=USIncExp2)
```

lead to equivalent results:

OLS-based CUSUM test

```
data: expenditure ~ income
S0 = 2.7578, p-value = 4.953e-07
```

`sctest` is a generic function which has methods not only for fluctuation tests, but all structural change tests (on historic data) introduced in this paper, i.e., also the F tests, which are described in the next section.

5 F tests

A completely different approach to investigate whether the null hypothesis of “no structural change” holds, is to use F test statistics. An important difference is that the alternative is specified: whereas the generalized fluctuation tests are suitable for various patterns of structural changes, the F tests are designed to test against a single shift alternative. Thus, the alternative can be formulated on the basis of the model (1)

$$\beta_i = \begin{cases} \beta_A & (1 \leq i \leq i_0) \\ \beta_B & (i_0 < i \leq n) \end{cases}, \quad (20)$$

where i_0 is some change point in the interval $(k, n - k)$. [Chow \(1960\)](#) was the first to suggest such a test on structural change if the (potential) change point i_0 is known. He proposed to fit a full OLS model, where the regression coefficients are estimated twice: once for the observations before the change point and once for those after i_0 . The resulting residuals $\hat{e} = (\hat{u}_A, \hat{u}_B)^\top$ can then be compared to the residuals \hat{u} from the restricted model, where the parameters are just fitted once for all observations. The test statistic

$$F_{i_0} = \frac{(\hat{u}^\top \hat{u} - \hat{e}^\top \hat{e})/k}{\hat{e}^\top \hat{e}/(n - 2k)}. \quad (21)$$

has an F distribution with k and $n - 2k$ degrees of freedom. The major drawback of this “Chow test” is that the change point has to be known in advance, but there are tests based upon F statistics (Chow statistics), that do not require a specification of a certain change point and which will be introduced in the following sections.

5.1 F statistics: function `Fstats`

A natural idea to extend the ideas from the Chow test is to calculate the F statistics for all potential change points or for all potential change points in an interval $[\underline{i}, \bar{i}]$ and to reject if any of those statistics get too large. Therefore the first step is to compute the F statistics F_i for $k < \underline{i} \leq i \leq \bar{i} < n - k$, which can be easily done using the function `Fstats`. Again the model to be tested is specified by a formula interface and the parameters \underline{i} and \bar{i} are respresented by `from` and `to`, respectively. Alternatively to indices these two parameters can also be specified by fractions of the sample; the default is to take `from = 0.15` and implicitly `to = 0.85`. To compute the F test statistics for all potential change points between January 1972 and December 1981 the appropriate command would be:

```
R> fs <- Fstats(expenditure ~ income, from=c(1972,1), to=c(1981,12), data=USIncExp2)
```

This returns an object of class “`Fstats`” which mainly contains a time series of F statistics. Analogously to the empirical fluctuation processes these objects can be printed, plotted and tested.

5.2 Boundaries and plotting

Just as for the empirical fluctuation processes it is possible to compute a constant boundary such that the (asymptotic) probability that one of the statistics F_i for $\underline{i} \leq i \leq \bar{i}$ exceeds this boundary is α under the null hypothesis of no structural change. Furthermore there are constant boundaries available for the mean of these statistics. The plotting of the process and its boundaries works in principle like introduced in the last section. So the command

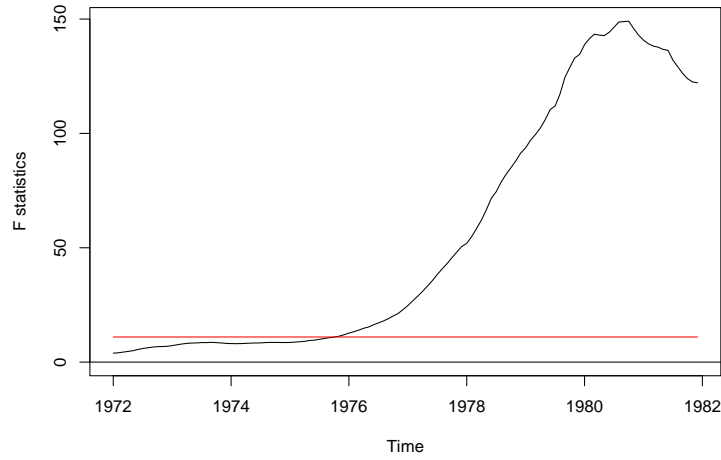


Figure 5: F statistics

```
R> plot(fs)
```

plots the process with its boundaries; the output can be seen in Figure 5. This shows a clear peak around 1981, which emphasizes the results from the analysis by empirical fluctuation processes and tests, respectively.

It is also possible to plot the p values instead of the F statistics themselves by

```
R> plot(fs, pval=TRUE)
```

which leads to equivalent results. Furthermore it is also possible to set up the boundaries for the average instead of the supremum by:

```
R> plot(fs, aveF=TRUE)
```

In this case another dashed line for the observed mean of the F statistics will be drawn.

5.3 Significance testing with F statistics

As already indicated in the previous section, there is more than one possibility to aggregate the series of F statistics into a test statistic. Hansen (1992) and Andrews (1993) respectively suggested three different test statistics and examined their asymptotic distribution:

$$\sup F = \sup_{i \leq i \leq \bar{i}} F_i, \quad (22)$$

$$\text{ave} F = \frac{1}{\bar{i} - \underline{i} + 1} \sum_{i=\underline{i}}^{\bar{i}} F_i, \quad (23)$$

$$\exp F = \log \left(\frac{1}{\bar{i} - \underline{i} + 1} \sum_{i=\underline{i}}^{\bar{i}} \exp(0.5 \cdot F_i) \right). \quad (24)$$

The $\sup F$ statistic in (22) and the $\text{ave} F$ statistic from (23) respectively reflect the testing procedures that have been described above. Either the null hypothesis is rejected when the maximal or the mean F statistic

gets too large. A third possibility is to reject when the exp^F statistic from (24) gets too large. The tests can be carried out in the same way as the fluctuation tests: either by supplying the fitted `Fstats` object or by a formula that describes the model to be tested. Hence both commands

```
R> sctest(fs, type="expF")
R> sctest(expenditure ~ income, type="expF", from = 25, to = 144, data = USIncExp2)
```

lead to equivalent output:

```
expF test

data: expenditure ~ income
exp.F = 71.0175, p-value = 1.692e-11
```

The p values are computed based on Hansen (1997).³

6 Monitoring with the generalized fluctuation test

In the previous sections we were concerned with the retrospective detection of structural changes in *given* data sets. Over the last years several structural change tests have been extended to online monitoring of linear regression models where new data arrive over time (Chu et al., 1996; Leisch et al., 2000). Such forward looking tests are closely related to sequential tests. When new observations arrive, estimates are computed sequentially from all available data (historical sample plus newly arrived data) and compared to the estimate based only on the historical sample. As in the retrospective case, the hypothesis of no structural change is rejected if the difference between these two estimates gets too large.

The standard linear regression model (1) is generalized to

$$y_i = x_i^\top \beta_i + u_i \quad (i = 1, \dots, n, n+1, \dots), \quad (25)$$

i.e., we expect new observations to arrive after time n (when the monitoring begins). The sample $\{(x_1, y_1), \dots, (x_n, y_n)\}$ will be called the *historic sample*, the corresponding time period $1, \dots, n$ the *history period*.

Currently monitoring has only been developed for recursive (Chu et al., 1996) and moving (Leisch et al., 2000) estimates tests. The respective limiting processes are—as in the retrospective case—the Brownian Bridge and increments of the Brownian Bridge. The empirical processes are rescaled to map the history period to the interval $[0,1]$ of the Brownian Bridge. For recursive estimates there exists a closed form solution for boundary functions, such that the limiting process Brownian Bridge stays within the boundaries on the interval $(1, \infty)$ with probability $1 - \alpha$. Note that the monitoring period consisting of all data arriving after the history period corresponds to the Brownian Bridge after time 1. For moving estimates, only the growth rate of boundaries can be derived analytically and critical values have to be simulated.

Consider that we want to monitor our income/expenditure data during the 1970's for structural change, using years 1969–1971 as history period. First we cut the historic sample from the complete data set and create an object of class "mefp":

```
R> USIncExp3 <- window(USIncExp, start=c(1969,1), end=c(1971,12))
R> me.mefp <- mefp(expenditure~income, type="ME", rescale=TRUE,
                  data=USIncExp3, alpha=0.05)
```

Because monitoring is a sequential test procedure, the significance level has to be specified *in advance*, i.e., when the object of class "mefp" is created. The "mefp" object can now be monitored repeatedly for structural changes.

Let us assume we get new observations for the year 1972. Calling function `monitor` on `me.mefp` automatically updates our monitoring object for the new observations and runs a sequential test for structural change on each new observation (no structural break is detected in 1972):

³The authors thank Bruce Hansen, who wrote the original code for computing p values for F statistics in GAUSS, for putting his code at disposal for porting to R.

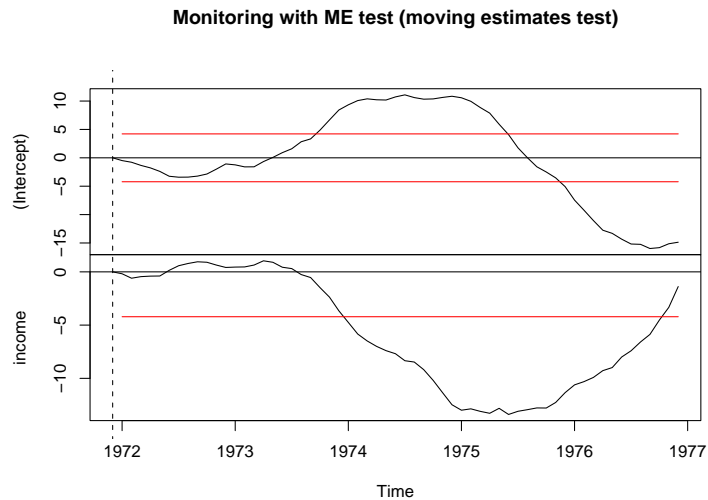


Figure 6: Monitoring structural change with bandwidth $h = 1$

```
R> USIncExp3 <- window(USIncExp, start=c(1969,1), end=c(1972,12))
R> me.mefp <- monitor(me.mefp)
```

Then new data for the years 1973-1976 arrive and we repeat the monitoring:

```
R> USIncExp3 <- window(USIncExp, start=c(1969,1), end=c(1976,12))
R> me.mefp <- monitor(me.mefp)
Break detected at observation # 58
R> me.mefp
Initial call:
  mefp.formula(obj = expenditure ~ income, data = USIncExp3, type = "ME",
               rescale = TRUE)
```

```
Last call:
  monitor(obj = me.mefp)
```

```
Significance level   : 0.05
Critical value       : 2.980014
History size         : 36
Last point evaluated : 96
Structural break at  : 58
```

```
Parameter estimate on history :
(Intercept)   income
  2.856749    0.770989
Last parameter estimate :
(Intercept)   income
-119.3429815  0.8606081
```

The software informs us that a structural break has been detected at observation #58, which corresponds to October 1973. Boundary and plotting methods for "mefp" objects work (almost) exactly as their "efp" counterparts, only the significance level α cannot be specified, because it is specified when the "mefp" object is created. The output of `plot(me.mefp)` can be seen in Figure 6. Instead of creating an "mefp" object using the formula interface like above, it could also be done re-using an existing "efp" object, e.g.:

```
R> USIncExp3 <- window(USIncExp, start=c(1969,1), end=c(1971,12))
R> me.efp <- efp(expenditure~income, type="ME", h=0.5, data=USIncExp3)
R> me.mefp <- mefp(me.efp, alpha=0.05)
```

If now again the new observations up to December 1976 arrive, we can monitor the data

```
R> USIncExp3 <- window(USIncExp, start=c(1969,1), end=c(1976,12))
R> me.mefp <- monitor(me.mefp)
Break detected at observation # 56
```

and discover the structural change even two observations earlier as we used the bandwidth $h=0.5$ instead of $h=1$. Due to this we have not one history estimate that is being compared with the new moving estimates, but we have a history process, which can be seen on the left in Figure 7. This plot can simply be generated by

```
R> plot(me.mefp)
```

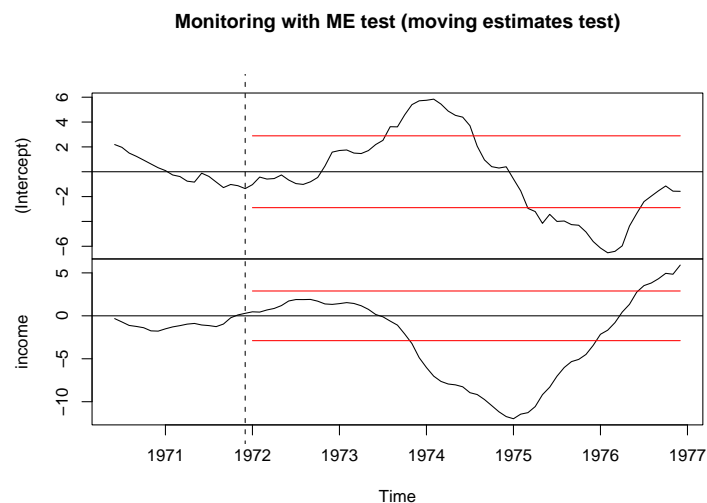


Figure 7: Monitoring structural change with bandwidth $h = 0.5$

7 Conclusions

In this paper, we introduce the `strucchange` package that implements methods for testing for structural change in linear regression relationships. It offers a unified framework to display information about structural changes flexibly and to assess their significance according to various tests.

Containing tests from the generalized fluctuation test framework as well as tests based on F statistics (Chow test statistics) the package extends standard significance testing procedures: There are methods to fit empirical fluctuation processes (CUSUM, MOSUM and estimates-based processes), compute an appropriate boundary, plot these results and finally carry out a formal significance test. Analogously a sequence of F statistics with the corresponding boundary can be computed, plotted and tested. Finally the methods for estimates-based fluctuation processes have extensions to monitor incoming data online.

Acknowledgements

This research of Achim Zeileis, Friedrich Leisch and Kurt Hornik was supported by the Austrian Science Foundation (FWF) under grant SFB#010 ('Adaptive Information Systems and Modeling in Economics and Management Science').

The work of Christian Kleiber was supported by the Deutsche Forschungsgemeinschaft, Sonderforschungsbereich 475.

References

- D. W. K. Andrews. Tests for parameter instability and structural change with unknown change point. *Econometrica*, 61:821–856, 1993.
- D. W. K. Andrews and W. Ploberger. Optimal tests when a nuisance parameter is present only under the alternative. *Econometrica*, 62:1383–1414, 1994.
- R. L. Brown, J. Durbin, and J. M. Evans. Techniques for testing the constancy of regression relationships over time. *Journal of the Royal Statistical Society*, B 37:149–163, 1975.
- G. C. Chow. Tests of equality between sets of coefficients in two linear regressions. *Econometrica*, 28: 591–605, 1960.
- C.-S. J. Chu, K. Hornik, and C.-M. Kuan. MOSUM tests for parameter constancy. *Biometrika*, 82:603–617, 1995a.
- C.-S. J. Chu, K. Hornik, and C.-M. Kuan. The moving-estimates test for parameter stability. *Econometric Theory*, 11:669–720, 1995b.
- C.-S. J. Chu, M. Stinchcombe, and H. White. Monitoring structural change. *Econometrica*, 64(5):1045–1065, 1996.
- B. E. Hansen. Tests for parameter instability in regressions with I(1) processes. *Journal of Business & Economic Statistics*, 10:321–335, 1992.
- B. E. Hansen. Approximate asymptotic p values for structural-change tests. *Journal of Business & Economic Statistics*, 15:60–67, 1997.
- W. Krämer, W. Ploberger, and R. Alt. Testing for structural change in dynamic models. *Econometrica*, 56 (6):1355–1369, 1988.
- C.-M. Kuan and M.-Y. Chen. Implementing the fluctuation and moving-estimates tests in dynamic econometric models. *Economics Letters*, 44:235–239, 1994.
- C.-M. Kuan and K. Hornik. The generalized fluctuation test: A unifying view. *Econometric Reviews*, 14: 135–161, 1995.
- F. Leisch, K. Hornik, and C.-M. Kuan. Monitoring structural changes with the generalized fluctuation test. *Econometric Theory*, 16:835–854, 2000.
- W. Ploberger and W. Krämer. The CUSUM test with OLS residuals. *Econometrica*, 60(2):271–285, 1992.
- A. Zeileis. p values and alternative boundaries for CUSUM tests. Working Paper 78, SFB "Adaptive Information Systems and Modelling in Economics and Management Science", December 2000a. URL <http://www.wu-wien.ac.at/am/wp00.htm#78>.
- A. Zeileis. p -Werte und alternative Schranken von CUSUM-Tests. Master's thesis, Fachbereich Statistik, Universität Dortmund, 2000b. URL <http://www.ci.tuwien.ac.at/~zeileis/papers/zeileis:2000.pdf>. In German.

A Implementation details for p values

An important and useful tool concerning significance tests are p values, especially for application in a software package. Their implementation is therefore crucial and in this section we will give more detail about the implementation in the `strucchange` package.

For the CUSUM tests with linear boundaries there are rather exact approximations to the asymptotic p value functions given in Zeileis (2000a). For the recursive estimates fluctuation test there is a series expansion, which is evaluated for the first hundred terms. For all other tests from the generalized fluctuation test framework the p values are computed by linear interpolation from tabulated critical values. For the Recursive CUSUM test with alternative boundaries p values from the interval $[0.001, 1]$ and $[0.0001, 1]$ for the OLS-based version respectively are approximated from tables given in Zeileis (2000b). The critical values for the Recursive MOSUM test for levels in $[0.01, 0.2]$ are taken from Chu et al. (1995a), while the critical values for the levels in $[0.01, 0.1]$ for the OLS-based MOSUM and the ME test are given in Chu et al. (1995b); the parameter h is in both cases interpolated for values in $[0.05, 0.5]$.

The p values for the $\text{sup}F$, $\text{ave}F$ and $\text{exp}F$ test are approximated based on Hansen (1997), who also wrote the original code in GAUSS, which we merely ported to R. The computation uses tabulated simulated regression coefficients.

B `strucchange` manual

`boundary.efp` *Boundary for Empirical Fluctuation Processes*

Description

Computes boundary for an object of class "efp"

Usage

```
boundary(x, alpha = 0.05, alt.boundary = FALSE, ...)
```

Arguments

<code>x</code>	an object of class "efp".
<code>alpha</code>	numeric from interval (0,1) indicating the confidence level for which the boundary of the corresponding test will be computed.
<code>alt.boundary</code>	logical. If set to TRUE alternative boundaries (instead of the standard linear boundaries) will be computed (for CUSUM processes only).
<code>...</code>	currently not used.

Value

an object of class "ts" with the same time properties as the time series in `x`

Author(s)

Achim Zeileis (zeileis@ci.tuwien.ac.at)

See Also

[efp](#), [plot.efp](#)

Examples

```
## Load dataset "nhtemp" with average yearly temperatures in New Haven
data(nhtemp)
## plot the data
plot(nhtemp)

## test the model null hypothesis that the average temperature remains constant
## over the years
## compute OLS-CUSUM fluctuation process
temp.cus <- efp(nhtemp ~ 1, type = "OLS-CUSUM")
## plot the process without boundaries
plot(temp.cus, alpha = 0.01, boundary = FALSE)
## add the boundaries in another colour
bound <- boundary(temp.cus, alpha = 0.01)
lines(bound, col=4)
lines(-bound, col=4)
```

`boundary.Fstats` *Boundary for F Statistics*

Description

Computes boundary for an object of class "Fstats"

Usage

```
boundary(x, alpha = 0.05, pval = FALSE, aveF = FALSE,
         asymptotic = FALSE, ...)
```

Arguments

<code>x</code>	an object of class "Fstats".
<code>alpha</code>	numeric from interval (0,1) indicating the confidence level for which the boundary of the supF test will be computed.
<code>pval</code>	logical. If set to TRUE a boundary for the corresponding p values will be computed.
<code>aveF</code>	logical. If set to TRUE the boundary of the aveF (instead of the supF) test will be computed. The resulting boundary then is a boundary for the mean of the F statistics rather than for the F statistics themselves.
<code>asymptotic</code>	logical. If set to TRUE the asymptotic (chi-square) distribution instead of the exact (F) distribution will be used to compute the p values (only if <code>pval</code> is TRUE).
<code>...</code>	currently not used.

Value

an object of class "ts" with the same time properties as the time series in `x`

Author(s)

Achim Zeileis (zeileis@ci.tuwien.ac.at)

See Also

[Fstats](#), [plot.Fstats](#)

Examples

```
## Load dataset "nhtemp" with average yearly temperatures in New Haven
data(nhtemp)
## plot the data
plot(nhtemp)

## test the model null hypothesis that the average temperature remains constant
## over the years for potential break points between 1941 (corresponds to
## from = 0.5) and 1962 (corresponds to to = 0.85)
## compute F statistics
fs <- Fstats(nhtemp ~ 1, from = 0.5, to = 0.85)
## plot the p values without boundary
plot(fs, pval = TRUE, alpha = 0.01)
## add the boundary in another colour
lines(boundary(fs, pval = TRUE, alpha = 0.01), col = 2)
```

`boundary.mefp`

Boundary Function for Monitoring of Structural Changes

Description

Computes boundary for an object of class "mefp"

Usage

```
boundary(x, ...)
```

Arguments

`x` an object of class "mefp".
`...` currently not used.

Value

an object of class "ts" with the same time properties as the monitored process

Author(s)

Friedrich Leisch

See Also

[mefp](#), [plot.mefp](#)

Examples

```
df1 <- data.frame(y=rnorm(300))
df1[150:300,"y"] <- df1[150:300,"y"]+1
me1 <- mefp(y~1, data=df1[1:50,,drop=FALSE], type="ME", h=1,
            alpha=0.05)
me2 <- monitor(me1, data=df1)

plot(me2, boundary=FALSE)
lines(boundary(me2), col="green", lty="44")
```

boundary

Boundary Function for Structural Change Tests

Description

A generic function computing boundaries for structural change tests

Usage

```
boundary(x, ...)
```

Arguments

`x` an object. Use [methods](#) to see which [class](#) has a method for `boundary`.
`...` additional arguments affecting the boundary.

Value

an object of class "ts" with the same time properties as the time series in `x`

Author(s)

Achim Zeileis (zeileis@ci.tuwien.ac.at)

See Also

[boundary.efp](#), [boundary.mefp](#), [boundary.Fstats](#)

covHC

Heteroskedasticity-Consistent Covariance Matrix Estimation

Description

Heteroskedasticity-consistent estimation of the covariance matrix of the coefficient estimates in a linear regression model.

Usage

```
covHC(formula, type = c("HC2", "const", "HC", "HC1", "HC3"), tol = 1e-10,  
data=list())
```

Arguments

formula	a symbolic description for the model to be tested.
type	a character string specifying the estimation type. For details see below.
tol	tolerance when <code>solve</code> is used
data	an optional data frame containing the variables in the model. By default the variables are taken from the environment which <code>covHC</code> is called from.

Details

When `type = "const"` constant variances are assumed and `covHC` gives the usual estimate of the covariance matrix of the coefficient estimates:

$$\hat{\sigma}^2(X^\top X)^{-1}$$

All other methods do not assume constant variances and are suitable in case of heteroskedasticity. "HC" gives White's estimator; for details see the references.

Value

A matrix containing the covariance matrix estimate.

References

MacKinnon J.G., White H. (1985), Some heteroskedasticity-consistent covariance matrix estimators with improved finite sample properties. *Journal of Econometrics* **29**, 305-325

See Also

[lm](#)

Examples

```
## generate linear regression relationship  
## with homoskedastic variances  
x <- sin(1:100)  
y <- 1 + x + rnorm(100)  
## compute usual covariance matrix of coefficient estimates  
covHC(y~x, type="const")  
  
sigma2 <- sum(residuals(lm(y~x))^2)/98  
sigma2 * solve(crossprod(cbind(1,x)))
```

Description

Computes an empirical fluctuation process according to a specified method from the generalized fluctuation test framework

Usage

```
efp(formula, data, type = <<see below>>, h = 0.15, dynamic = FALSE,
    rescale = TRUE, tol = 1e-7)
```

Arguments

formula	a symbolic description for the model to be tested.
data	an optional data frame containing the variables in the model. By default the variables are taken from the environment which <code>efp</code> is called from.
type	specifies which type of fluctuation process will be computed. For details see below.
h	a numeric from interval (0,1) specifying the bandwidth. determines the size of the data window relative to sample size (for MOSUM and ME processes only).
dynamic	logical. If TRUE the lagged observations are included as a regressor.
rescale	logical. If TRUE the estimates will be standardized by the regressor matrix of the corresponding subsample according to Kuan & Chen (1994); if FALSE the whole regressor matrix will be used. (only if <code>type</code> is either "fluctuation" or "ME")
tol	tolerance when <code>solve</code> is used

Details

If `type` is one of "Rec-CUSUM", "OLS-CUSUM", "Rec-MOSUM" or "OLS-MOSUM" the function `efp` will return a one-dimensional empirical process of sums of residuals. Either it will be based on recursive residuals or on OLS residuals and the process will contain CUMulative SUMs or MOving SUMs of residuals in a certain data window. For the MOSUM and ME processes all estimations are done for the observations in a moving data window, whose size is determined by `h` and which is shifted over the whole sample.

If there is a single structural change point t^* , the standard CUSUM path starts to depart from its mean 0 at t^* . The OLS-based CUSUM path will have its peak around t^* . The MOSUM path should have a strong change at t^* .

If `type` is either "fluctuation" or "ME" a k -dimensional process will be returned, if k is the number of regressors in the model, as it is based on recursive OLS estimates of the regression coefficients or moving OLS estimates respectively.

Both paths should have a peak around t^* if there is a single structural shift.

Value

`efp` returns an object of class "efp" which inherits from the class "ts" or "mts" respectively, to which a string with the `type` of the process, the number of regressors (`nreg`), the bandwidth (`h`) and the function call (`call`) are added as attributes. The function `plot` has a method to plot the empirical fluctuation process; with `sctest` the corresponding test on structural change can be performed.

Author(s)

Achim Zeileis <zeileis@ci.tuwien.ac.at>

References

- Brown R.L., Durbin J., Evans J.M. (1975), Techniques for testing constancy of regression relationships over time, *Journal of the Royal Statistical Society, B*, **37**, 149-163.
- Chu C.-S., Hornik K., Kuan C.-M. (1995), MOSUM tests for parameter constancy, *Biometrika*, **82**, 603-617.
- Chu C.-S., Hornik K., Kuan C.-M. (1995), The moving-estimates test for parameter stability, *Econometric Theory*, **11**, 669-720.
- Krämer W., Ploberger W., Alt R. (1988), Testing for structural change in dynamic models, *Econometrica*, **56**, 1355-1369.
- Kuan C.-M., Hornik K. (1995), The generalized fluctuation test: A unifying view, *Econometric Reviews*, **14**, 135 - 161.
- Kuan C.-M., Chen (1994), Implementing the fluctuation and moving estimates tests in dynamic econometric models, *Economics Letters*, **44**, 235-239.
- Ploberger W., Krämer W. (1992), The CUSUM test with OLS residuals, *Econometrica*, **60**, 271-285.

See Also

[plot.efp](#), [print.efp](#), [sctest.efp](#), [boundary.efp](#)

Examples

```
## Load dataset "nhtemp" with average yearly temperatures in New Haven
data(nhtemp)
## plot the data
plot(nhtemp)

## test the model null hypothesis that the average temperature remains constant
## over the years
## compute OLS-CUSUM fluctuation process
temp.cus <- efp(nhtemp ~ 1, type = "OLS-CUSUM")
## plot the process with alternative boundaries
plot(temp.cus, alpha = 0.01, alt.boundary = TRUE)
## and calculate the test statistic
sctest(temp.cus)

## Load dataset "USIncExp" with income and expenditure in the US
## and choose a suitable subset
data(USIncExp)
USIncExp2 <- window(USIncExp, start=c(1970,1), end=c(1989,12))

## test the null hypothesis that the way the income is spent in expenditure
## does not change over time
## compute moving estimates fluctuation process
me <- efp(expenditure~income, type="ME", data=USIncExp2, h=0.2)
## plot the two dimensional fluctuation process with boundaries
plot(me, functional=NULL)
## and perform the corresponding test
sctest(me)
```

Description

Computes a series of F statistics for a specified data window.

Usage

```
Fstats(formula, from = 0.15, to = NULL, data,
       cov.type = c("const", "HC", "HC1"), tol=1e-7)
```

Arguments

<code>formula</code>	a symbolic description for the model to be tested
<code>from, to</code>	numeric. If <code>from</code> is smaller than 1 they are interpreted as percentages of data and by default <code>to</code> is taken to be <code>1 - from</code> . F statistics will be calculated for the observations <code>(n*from):(n*to)</code> , when <code>n</code> is the number of observations in the model. If <code>from</code> is greater than 1 it is interpreted to be the index and <code>to</code> defaults to <code>n - from</code> . If <code>from</code> is a vector with two elements, then <code>from</code> and <code>to</code> are interpreted as time specifications like in ts , see also the examples.
<code>data</code>	an optional data frame containing the variables in the model. By default the variables are taken from the environment which <code>Fstats</code> is called from.
<code>cov.type</code>	a string indicating which type of covariance matrix estimator should be used. Constant homoskedastic variances are assumed if set to <code>"const"</code> and White's heteroskedasticity consistent estimator is used if set to <code>"HC"</code> . And <code>"HC1"</code> stands for a standardized estimator of <code>"HC"</code> , see also covHC .
<code>tol</code>	tolerance when <code>solve</code> is used.

Details

For every potential change point in `from:to` a F statistic (Chow test statistic) is computed. For this an OLS model is fitted for the observations before and after the potential change point, i.e. `2k` parameters have to be estimated, and the error sum of squares is computed (ESS). Another OLS model for all observations with a restricted sum of squares (RSS) is computed, hence `k` parameters have to be estimated here. If `n` is the number of observations and `k` the number of regressors in the model, the formula is:

$$F = \frac{(RSS - ESS)}{ESS/(n - 2k)}$$

Value

`Fstats` returns an object of class `"Fstats"`, which contains mainly a time series of F statistics. The function [plot](#) has a method to plot the F statistics or the corresponding p values; with `sctest` a supF-, aveF- or expF-test on structural change can be performed.

Author(s)

Achim Zeileis (zeileis@ci.tuwien.ac.at)

References

Andrews D.W.K. (1993), Tests for parameter instability and structural change with unknown change point, *Econometrica*, **61**, 821-856.

Hansen B. (1992), Tests for parameter instability in regressions with I(1) processes, *Journal of Business & Economic Statistics*, **10**, 321-335.

Hansen B. (1997), Approximate asymptotic p values for structural-change tests, *Journal of Business & Economic Statistics*, **15**, 60-67.

See Also

[plot.Fstats](#), [sctest.Fstats](#), [boundary.Fstats](#)

Examples

```
## Load dataset "nhtemp" with average yearly temperatures in New Haven
data(nhtemp)
## plot the data
plot(nhtemp)

## test the model null hypothesis that the average temperature remains constant
## over the years for potential break points between 1941 (corresponds to from =
## 0.5) and 1962 (corresponds to to = 0.85)
## compute F statistics
fs <- Fstats(nhtemp ~ 1, from = 0.5, to = 0.85)
## this gives the same result
fs <- Fstats(nhtemp ~ 1, from = c(1941,1), to = c(1962,1))
## plot the F statistics
plot(fs, alpha = 0.01)
## and the corresponding p values
plot(fs, pval = TRUE, alpha = 0.01)
## perform the aveF test
sctest(fs, type = "aveF")
```

mefp

Monitoring of Empirical Fluctuation Processes

Description

Online monitoring of structural breaks in a linear regression model. A parameter estimate based on a historical sample is compared with estimates based on newly arriving data; a sequential test on the difference between the two parameter estimates signals structural breaks.

Usage

```
mefp(obj, ...)
```

```
mefp(formula, type = c("OLS-CUSUM", "OLS-MOSUM", "ME",
  "fluctuation"), data, h=1, alpha=0.05, functional = c("max", "range"),
  period=10, tolerance=.Machine$double.eps^0.5,
  CritvalTable=NULL, rescale=NULL, border=NULL, ...)
```

```
mefp(obj, alpha=0.05, functional = c("max", "range"),
      period=10, tolerance=.Machine$double.eps^0.5,
      CritvalTable=NULL, rescale=NULL, border=NULL, ...)
```

```
monitor(obj, data=NULL, verbose=TRUE)
```

Arguments

<code>formula</code>	a symbolic description for the model to be tested.
<code>data</code>	an optional data frame containing the variables in the model. By default the variables are taken from the environment which <code>mefp</code> is called from.
<code>type</code>	specifies which type of fluctuation process will be computed.
<code>h</code>	(only used for ME processes). A numeric scalar from interval (0,1) specifying the size of the data window relative to the sample size.
<code>obj</code>	Object of class " <code>mefp</code> " (for <code>mefp</code>) or " <code>mefp</code> " (for <code>monitor</code>).
<code>alpha</code>	Significance level of the test, i.e., probability of type I error.
<code>functional</code>	Determines if maximum or range of parameter differences is used as statistic.
<code>period</code>	(only used for ME processes). Maximum time (relative to the history period) that will be monitored. Default is 10 times the history period.
<code>tolerance</code>	Tolerance for numeric == comparisons.
<code>CritvalTable</code>	Table of critical values, this table is interpolated to get critical values for arbitrary alphas. The default depends on the <code>type</code> of fluctuation process (pre-computed tables are available for all types). <i>This argument is under development.</i>
<code>rescale</code>	If TRUE the estimates will be standardized by the regressor matrix of the corresponding subsample similar to Kuan & Chen (1994); if FALSE the historic regressor matrix will be used. The default is to rescale the monitoring processes of type "ME" but not of "fluctuation".
<code>border</code>	An optional user-specified border function for the empirical process. <i>This argument is under development.</i>
<code>verbose</code>	If TRUE, signal breaks by text output.
<code>...</code>	Currently not used.

Details

`mefp` creates an object of class "`mefp`" either from a model formula or from an object of class "`mefp`". In addition to the arguments of `mefp`, the type of statistic and a significance level for the monitoring must be specified. The monitoring itself is performed by `monitor`, which can be called arbitrarily often on objects of class "`mefp`". If new data have arrived, then the empirical fluctuation process is computed for the new data. If the process crosses the boundaries corresponding to the significance level `alpha`, a structural break is detected (and signaled).

The typical usage is to initialize the monitoring by creation of an object of class "`mefp`" either using a formula or an "`mefp`" object. Data available at this stage are considered the *history sample*, which is kept fixed during the complete monitoring process, and may not contain any structural changes.

Subsequent calls to `monitor` perform a sequential test of the null hypothesis of no structural change in new data against the general alternative of changes in one or more of the coefficients of the regression model.

Author(s)

Friedrich Leisch

References

Friedrich Leisch, Kurt Hornik, and Chung-Ming Kuan. Monitoring structural changes with the generalized fluctuation test. *Econometric Theory*, 16:835-854, 2000.

See Also

[plot.mefp](#), [boundary.mefp](#)

Examples

```
df1 <- data.frame(y=rnorm(300))
df1[150:300,"y"] <- df1[150:300,"y"]+1

## use the first 50 observations as history period
e1 <- efp(y~1, data=df1[1:50,,drop=FALSE], type="ME", h=1)
me1 <- mefp(e1, alpha=0.05)

## the same in one function call
me1 <- mefp(y~1, data=df1[1:50,,drop=FALSE], type="ME", h=1,
           alpha=0.05)

## monitor the 50 next observations
me2 <- monitor(me1, data=df1[1:100,,drop=FALSE])
plot(me2)

# and now monitor on all data
me3 <- monitor(me2, data=df1)
plot(me3)

## Load dataset "USIncExp" with income and expenditure in the US
## and choose a suitable subset for the history period
data(USIncExp)
USIncExp3 <- window(USIncExp, start=c(1969,1), end=c(1971,12))
## initialize the monitoring with the formula interface
me.mefp <- mefp(expenditure~income, type="ME", rescale=TRUE,
               data=USIncExp3, alpha=0.05)

## monitor the new observations for the year 1972
USIncExp3 <- window(USIncExp, start=c(1969,1), end=c(1972,12))
me.mefp <- monitor(me.mefp)

## monitor the new data for the years 1973-1976
USIncExp3 <- window(USIncExp, start=c(1969,1), end=c(1976,12))
me.mefp <- monitor(me.mefp)
plot(me.mefp, functional = NULL)
```

plot.efp

Plot Empirical Fluctuation Process

Description

Plotting method for objects of class "efp"

Usage

```
plot(x, alpha = 0.05, alt.boundary = FALSE, boundary = TRUE,  
     functional = "max", main = NULL, ylim = NULL,  
     ylab = "empirical fluctuation process", ...)
```

Arguments

`x` an object of class "efp".

`alpha` numeric from interval (0,1) indicating the confidence level for which the boundary of the corresponding test will be computed.

`alt.boundary` logical. If set to TRUE alternative boundaries (instead of the standard linear boundaries) will be plotted (for CUSUM processes only).

`boundary` logical. If set to FALSE the boundary will be computed but not plotted.

`functional` indicates which functional should be applied to the estimates based processes ("fluctuation" and "ME"). If set to NULL a multiple process is plotted.

`main, ylim, ylab, ...` high-level `plot` function parameters.

Details

Alternative boundaries that are proportional to the standard deviation of the corresponding limiting process are available for the CUSUM-type processes.

Value

`efp` returns an object of class "efp" which inherits from the class "ts" or "mts" respectively. The function `plot` has a method to plot the empirical fluctuation process; with `sctest` the corresponding test on structural change can be performed.

Author(s)

Achim Zeileis (zeileis@ci.tuwien.ac.at)

References

- Brown R.L., Durbin J., Evans J.M. (1975), Techniques for testing constancy of regression relationships over time, *Journal of the Royal Statistical Society, B*, **37**, 149-163.
- Chu C.-S., Hornik K., Kuan C.-M. (1995), MOSUM tests for parameter constancy, *Biometrika*, **82**, 603-617.
- Chu C.-S., Hornik K., Kuan C.-M. (1995), The moving-estimates test for parameter stability, *Econometric Theory*, **11**, 669-720.
- Krämer W., Ploberger W., Alt R. (1988), Testing for structural change in dynamic models, *Econometrica*, **56**, 1355-1369.
- Kuan C.-M., Hornik K. (1995), The generalized fluctuation test: A unifying view, *Econometric Reviews*, **14**, 135 - 161.
- Kuan C.-M., Chen (1994), Implementing the fluctuation and moving estimates tests in dynamic econometric models, *Economics Letters*, **44**, 235-239.
- Ploberger W., Krämer W. (1992), The CUSUM test with OLS residuals, *Econometrica*, **60**, 271-285.

Zeileis A. (2000), p Values and Alternative Boundaries for CUSUM Tests, Working Paper 78, SFB "Adaptive Information Systems and Modelling in Economics and Management Science", Vienna University of Economics, <http://www.wu-wien.ac.at/am/wp00.htm#78>.

See Also

[efp](#), [boundary.efp](#), [sctest.efp](#)

Examples

```
## Load dataset "nhtemp" with average yearly temperatures in New Haven
data(nhtemp)
## plot the data
plot(nhtemp)

## test the model null hypothesis that the average temperature remains constant
## over the years
## compute Rec-CUSUM fluctuation process
temp.cus <- efp(nhtemp ~ 1)
## plot the process
plot(temp.cus, alpha = 0.01)
## and calculate the test statistic
sctest(temp.cus)

## compute (recursive estimates) fluctuation process
## with an additional linear trend regressor
lin.trend <- 1:60
temp.me <- efp(nhtemp ~ lin.trend, type = "fluctuation")
## plot the bivariate process
plot(temp.me, functional = NULL)
## and perform the corresponding test
sctest(temp.me)
```

`plot.Fstats` *Plot F Statistics*

Description

Plotting method for objects of class "Fstats"

Usage

```
plot(x, pval = FALSE, asymptotic = FALSE, alpha = 0.05, boundary = TRUE,
     aveF = FALSE, xlab = "Time", ylab = NULL, ylim = NULL, ...)
```

Arguments

<code>x</code>	an object of class "Fstats".
<code>pval</code>	logical. If set to TRUE the corresponding p values instead of the original F statistics will be plotted.
<code>asymptotic</code>	logical. If set to TRUE the asymptotic (chi-square) distribution instead of the exact (F) distribution will be used to compute the p values (only if <code>pval</code> is TRUE).

alpha numeric from interval (0,1) indicating the confidence level for which the boundary of the supF test will be computed.

boundary logical. If set to FALSE the boundary will be computed but not plotted.

aveF logical. If set to TRUE the boundary of the aveF test will be plotted. As this is a boundary for the mean of the F statistics rather than for the F statistics themselves a dashed line for the mean of the F statistics will also be plotted.

xlab, ylab, ylim, ...
 high-level `plot` function parameters.

Author(s)

Achim Zeileis (zeileis@ci.tuwien.ac.at)

References

- Andrews D.W.K. (1993), Tests for parameter instability and structural change with unknown change point, *Econometrica*, **61**, 821-856.
- Hansen B. (1992), Tests for parameter instability in regressions with I(1) processes, *Journal of Business & Economic Statistics*, **10**, 321-335.
- Hansen B. (1997), Approximate asymptotic p values for structural-change tests, *Journal of Business & Economic Statistics*, **15**, 60-67.

See Also

[Fstats](#), [boundary.Fstats](#), [sctest.Fstats](#)

Examples

```
## Load dataset "nhtemp" with average yearly temperatures in New Haven
data(nhtemp)
## plot the data
plot(nhtemp)

## test the model null hypothesis that the average temperature remains constant
## over the years for potential break points between 1941 (corresponds to
## from = 0.5) and 1962 (corresponds to to = 0.85)
## compute F statistics
fs <- Fstats(nhtemp ~ 1, from = 0.5, to = 0.85)
## plot the F statistics
plot(fs, alpha = 0.01)
## and the corresponding p values
plot(fs, pval = TRUE, alpha = 0.01)
## perform the aveF test
sctest(fs, type = "aveF")
```

Description

This is a method of the generic `plot` function for for "mefp" objects as returned by `mefp` or `monitor`. It plots the empirical fluctuation process (or a functional thereof) as a time series plot, and includes boundaries corresponding to the significance level of the monitoring procedure.

Usage

```
plot(x, boundary=TRUE, functional="max", main=NULL,
      ylab="empirical fluctuation process", ylim=NULL, ...)
```

Arguments

`x` an object of class "mefp".
`boundary` if FALSE, plotting of boundaries is suppressed.
`functional` indicates which functional should be applied to a multivariate empirical process. If set to NULL all dimensions of the process (one process per coefficient in the linear model) are plotted.
`main, ylab, ylim, ...` high-level `plot` function parameters.

Author(s)

Friedrich Leisch

See Also

[mefp](#)

Examples

```
df1 <- data.frame(y=rnorm(300))
df1[150:300,"y"] <- df1[150:300,"y"]+1
me1 <- mefp(y~1, data=df1[1:50,,drop=FALSE], type="ME", h=1,
            alpha=0.05)
me2 <- monitor(me1, data=df1)

plot(me2)
```

`root.matrix`

Root of a Matrix

Description

Computes the root of a symmetric and positive semidefinite matrix.

Usage

```
root.matrix(X)
```

Arguments

`X` a symmetric and positive semidefinite matrix

Value

a symmetric matrix of same dimensions as `X`

Author(s)

Achim Zeileis (zeileis@ci.tuwien.ac.at)

Examples

```
X <- matrix(c(1,2,2,8), ncol=2)
test <- root.matrix(X)
## control results
X
test %*% test
```

`sctest.efp`

Generalized Fluctuation Tests

Description

Performs a generalized fluctuation test.

Usage

```
sctest(x, alt.boundary = FALSE, functional = c("max", "range"),
...)
```

Arguments

`x` an object of class "efp".

`alt.boundary` logical. If set to TRUE alternative boundaries (instead of the standard linear boundaries) will be used (for CUSUM processes only).

`functional` indicates which functional should be applied to the estimates based processes ("fluctuation" and "ME").

... currently not used.

Details

The critical values for the MOSUM tests and the ME test are just tabulated for confidence levels between 0.1 and 0.01, thus the p value approximations will be poor for other p values.

Value

an object of class "htest" containing:

statistic	the test statistic
p.value	the corresponding p value
method	a character string with the method used
data.name	a character string with the data name

Author(s)

Achim Zeileis (zeileis@ci.tuwien.ac.at)

References

- Brown R.L., Durbin J., Evans J.M. (1975), Techniques for testing constancy of regression relationships over time, *Journal of the Royal Statistical Society*, B, **37**, 149-163.
- Chu C.-S., Hornik K., Kuan C.-M. (1995), MOSUM tests for parameter constancy, *Biometrika*, **82**, 603-617.
- Chu C.-S., Hornik K., Kuan C.-M. (1995), The moving-estimates test for parameter stability, *Econometric Theory*, **11**, 669-720.
- Krämer W., Ploberger W., Alt R. (1988), Testing for structural change in dynamic models, *Econometrica*, **56**, 1355-1369.
- Kuan C.-M., Hornik K. (1995), The generalized fluctuation test: A unifying view, *Econometric Reviews*, **14**, 135 - 161.
- Kuan C.-M., Chen (1994), Implementing the fluctuation and moving estimates tests in dynamic econometric models, *Economics Letters*, **44**, 235-239.
- Ploberger W., Krämer W. (1992), The CUSUM Test with OLS Residuals, *Econometrica*, **60**, 271-285.
- Zeileis A. (2000), p Values and Alternative Boundaries for CUSUM Tests, Working Paper 78, SFB "Adaptive Information Systems and Modelling in Economics and Management Science", Vienna University of Economics, <http://www.wu-wien.ac.at/am/wp00.htm#78>.

See Also

[efp](#), [plot.efp](#)

Examples

```
## Load dataset "nhtemp" with average yearly temperatures in New Haven
data(nhtemp)
## plot the data
plot(nhtemp)

## test the model null hypothesis that the average temperature remains constant
## over the years compute OLS-CUSUM fluctuation process
temp.cus <- efp(nhtemp ~ 1, type = "OLS-CUSUM")
## plot the process with alternative boundaries
plot(temp.cus, alpha = 0.01, alt.boundary = TRUE)
## and calculate the test statistic
sctest(temp.cus)

## compute moving estimates fluctuation process
```

```
temp.me <- efp(nhtemp ~ 1, type = "ME", h = 0.2)
## plot the process with functional = "max"
plot(temp.me)
## and perform the corresponding test
sctest(temp.me)
```

sctest.Fstats *supF-, aveF- and expF-Test*

Description

Performs the supF-, aveF- or expF-test

Usage

```
sctest(x, type = c("supF", "aveF", "expF"),
       asymptotic = FALSE, ...)
```

Arguments

<code>x</code>	an object of class "Fstats".
<code>type</code>	a character string specifying which test will be performed.
<code>asymptotic</code>	logical. Only necessary if <code>x</code> contains just a single F statistic and <code>type</code> is "supF" or "aveF". If then set to TRUE the asymptotic (chi-square) distribution instead of the exact (F) distribution will be used to compute the p value.
<code>...</code>	currently not used.

Details

If `x` contains just a single F statistic and `type` is "supF" or "aveF" the Chow test will be performed. The original GAUSS code for computing the p values of the supF-, aveF- and expF-test was written by Bruce Hansen and is available from <http://www.ssc.wisc.edu/~bhansen/>. R port by Achim Zeileis.

Value

an object of class "htest" containing:

<code>statistic</code>	the test statistic
<code>p.value</code>	the corresponding p value
<code>method</code>	a character string with the method used
<code>data.name</code>	a character string with the data name

References

Andrews D.W.K. (1993), Tests for parameter instability and structural change with unknown change point, *Econometrica*, **61**, 821-856.

Hansen B. (1992), Tests for parameter instability in regressions with I(1) processes, *Journal of Business & Economic Statistics*, **10**, 321-335.

Hansen B. (1997), Approximate asymptotic p values for structural-change tests, *Journal of Business & Economic Statistics*, **15**, 60-67.

See Also

[Fstats](#), [plot.Fstats](#)

Examples

```
## Load dataset "nhtemp" with average yearly temperatures in New Haven
data(nhtemp)
## plot the data
plot(nhtemp)

## test the model null hypothesis that the average temperature remains constant
## over the years for potential break points between 1941 (corresponds to
## from = 0.5) and 1962 (corresponds to to = 0.85)
## compute F statistics
fs <- Fstats(nhtemp ~ 1, from = 0.5, to = 0.85)
## plot the F statistics
plot(fs, alpha = 0.01)
## and the corresponding p values
plot(fs, pval = TRUE, alpha = 0.01)
## perform the aveF test
sctest(fs, type = "aveF")
```

`sctest.formula` *Structural Change Tests*

Description

Performs tests for structural change.

Usage

```
sctest(formula, type = <<see below>>, h = 0.15,
       alt.boundary = FALSE, functional = c("max", "range"),
       from = 0.15, to = NULL, point = 0.5, asymptotic = FALSE, data, ...)
```

Arguments

<code>formula</code>	a formula describing the model to be tested.
<code>type</code>	a character string specifying the structural change test that ist to be performed. Besides the tests types described in efp and sctest.Fstats the Chow test is can be performed by setting type to "Chow".
<code>h</code>	numeric from interval (0,1) specifying the bandwidth. Determines the size of the data window relative to sample size (for MOSUM and ME tests only).
<code>dynamic</code>	logical. If TRUE the lagged observations are included as a regressor (for generalized fluctuation tests only).
<code>tol</code>	tolerance when solve is used
<code>alt.boundary</code>	logical. If set to TRUE alternative boundaries (instead of the standard linear boundaries) will be used (for CUSUM processes only).
<code>functional</code>	indicates which functional should be applied to the estimates based processes ("fluctuation" and "ME").

<code>from, to</code>	numerics. If <code>from</code> is smaller than 1 they are interpreted as percentages of data and by default <code>to</code> is taken to be the 1 - <code>from</code> . F statistics will be calculated for the observations $(n \cdot \text{from}) : (n \cdot \text{to})$, when <code>n</code> is the number of observations in the model. If <code>from</code> is greater than 1 it is interpreted to be the index and <code>to</code> defaults to <code>n - from</code> . (for F tests only)
<code>point</code>	parameter of the Chow test for the potential change point. Interpreted analogous to the <code>from</code> parameter. By default taken to be <code>floor(n * 0.5)</code> if <code>n</code> is the number of observations in the model.
<code>asymptotic</code>	logical. If <code>TRUE</code> the asymptotic (chi-square) distribution instead of the exact (F) distribution will be used to compute the p value (for Chow test only).
<code>data</code>	an optional data frame containing the variables in the model. By default the variables are taken from the environment which <code>sctest</code> is called from.
<code>...</code>	further arguments passed to <code>efp</code> or <code>Fstats</code> .

Details

`sctest.formula` is mainly a wrapper for `sctest.efp` and `sctest.Fstats` as it fits an empirical fluctuation process first or computes the F statistics respectively and subsequently performs the corresponding test. The Chow test is available explicitly here.

Value

an object of class "htest" containing:

<code>statistic</code>	the test statistic
<code>p.value</code>	the corresponding p value
<code>method</code>	a character string with the method used
<code>data.name</code>	a character string with the data name

Author(s)

Achim Zeileis (zeileis@ci.tuwien.ac.at)

See Also

[sctest.efp](#), [sctest.Fstats](#)

Examples

```
## Load dataset "nhtemp" with average yearly temperatures in New Haven
data(nhtemp)
## plot the data
plot(nhtemp)

## test the model null hypothesis that the average temperature remains constant
## over the years with the Standard CUSUM test
sctest(nhtemp ~ 1)
## with the Chow test (under the alternative that there is a change 1941)
sctest(nhtemp ~ 1, type = "Chow", point = c(1941,1))
```

strucchange.internal

Internal strucchange objects

Description

These are not to be called by the user.

Author(s)

Achim Zeileis, Friedrich Leisch

USIncExp

Income and Expenditures in the US

Description

Data set containing the monthly personal income and personal consumption expenditures (in billion US dollars) between January 1959 and February 2001, which is seasonally adjusted at annual rates.

Source

<http://www.economagic.com/>