

Testing for Structural Change

Theory, Implementation and Applications

DISSERTATION

zur Erlangung des akademischen Grades
eines Doktors der Naturwissenschaften
der Universität Dortmund

Dem Fachbereich Statistik
der Universität Dortmund
vorgelegt von

Achim Zeileis

Dortmund 2003

Prüfungskommission: Prof. Dr. Claus Weihs (Vorsitzender)
Prof. Dr. Walter Krämer (Gutachter)
Prof. Dr. Kurt Hornik (Gutachter)
Prof. Dr. Götz Trenkler (Gutachter)
Dr. Lars Tschiersch (Beisitzer)

Tag der mündlichen Prüfung: 14. November 2003

Danksagung

Nach drei Jahren Arbeit an der Technischen Universität Wien im Rahmen des vom FWF (Fonds zur Förderung der wissenschaftlichen Forschung) geförderten SFB 010 "Adaptive Information Systems and Modelling in Economics and Management Science" habe ich nun die vorliegende Dissertation abschließen können. Auch wenn auf dem Umschlag nur mein Name steht, so wäre die Aufnahme, Durchführung und erfolgreiche Beendigung ohne die Hilfe und Unterstützung vieler Kollegen und Freunde nicht möglich gewesen.

An erster Stelle gilt mein Dank all denen, die durch gemeinsame Forschung die wissenschaftliche Basis zu dieser Arbeit gelegt haben: Insbesondere möchte ich Kurt Hornik danken, der diese Dissertation betreut hat, mir als wissenschaftlicher Mentor immer zur Seite stand und mir aber auch die Freiheit gelassen hat, meine Forschung nach eigenen Interessen auszurichten. Weiter danke ich Walter Krämer, der mein Interesse für das Themengebiet dieser Arbeit überhaupt geweckt hat und mich von Beginn meiner Tätigkeit als studentische Hilfskraft an über die Diplomarbeit bis hin zum Abschluß dieser Dissertation unterstützt und mir im Institut für Wirtschafts- und Sozialstatistik an der Universität Dortmund immer eine Anlaufstelle geboten hat. Diese Beziehung von Wien nach Dortmund mit (wissenschaftlichem) Leben gefüllt hat besonders Christian Kleiber, dem ich für

sein Engagement und seine in der Regel kritischen aber immer konstruktiven Beiträge zu einer sehr angenehmen und produktiven Forschungsoperation danke. Bei Friedrich Leisch möchte ich mich dafür bedanken, daß er mir geholfen hat, mich im Alltag des Universitätsgeschäfts zurechtzufinden. Außerdem bin ich ihm und Kurt Hornik zu Dank verpflichtet, da sie mir nicht nur viel beigebracht haben über wissenschaftliches Arbeiten im Allgemeinen und über die Umsetzung von statistischen Ideen in informatische Konzepte, sondern mir auch immer eine Plattform zu selbständigem Arbeiten geboten haben. Besonders danken möchte ich auch Torsten Hothorn, der in gewisser Weise an der Ausrichtung dieser Arbeit Schuld ist, da ich ohne ihn überhaupt nicht in Wien gelandet wäre. Zudem war er auch immer ein guter Diskussionspartner, sowohl für gemeinsame wissenschaftliche Projekte als auch für darüber hinausgehende allgemeinere Fragen.

Für eine angenehme wissenschaftliche Zusammenarbeit sowie streng nicht-wissenschaftliche Aktivitäten abseits des Berufs möchte ich mich bei meinen Wiener Kollegen und Freunden bedanken: Evgenia Dimitriadou, Alexandros Karatzoglou und David Meyer haben über die letzten drei Jahre hinweg unser Büro zu einem Ort gemacht, an dem ich gerne mehr als nur die Arbeitszeit verbracht habe. Auch Bettina Grün, die erst Anfang des Jahres, dazugekommen ist, hat sich sehr gut in diese Atmosphäre eingefügt und sie bereichert. Ein spezieller Dank geht an Ortrun Veichtlbauer, die nicht nur für eine spannende interdisziplinäre Kooperation verantwortlich ist, sondern auch mehr als einmal mein seelisches Gleichgewicht zurecht gerückt hat, und ohne dies hätte ich es vielleicht nicht so lang in Wien ausgehalten.

Alle wissenschaftlichen Leistungen wären aber nicht viel wert, ohne die

Menschen, die mir im Leben außerhalb der Universität Kraft und Rückhalt geben. Deshalb möchte ich all meinen Freunden danken, die mich auch in Zeiten extremer Büroaufenthaltsdauern meinerseits nicht vergessen haben und für mich da waren, um mich an die wichtigeren Dinge im Leben zu erinnern. Das Fundament für all das habe ich aber meiner Familie zu verdanken, meinen Eltern und meinem Bruder, die mich immer in jeder Hinsicht unterstützen und ohne die ich nicht da wäre, wo ich heute bin.

Achim Zeileis

Wien, November 2003

Contents

| | |
|--|-----------|
| Abstract | 1 |
| Zusammenfassung | 3 |
| 1 Introduction | 5 |
| 1.1 Tests for Structural Change | 7 |
| 1.2 Diagnostic Checking vs. Data Exploration | 9 |
| 1.3 Visualization and Significance Testing | 11 |
| 1.4 Segmentation of Regression Models | 12 |
| 1.5 Overview | 13 |
| 2 Generalized M-Fluctuation Tests for Parameter Instability | 17 |
| 2.1 Introduction | 17 |
| 2.2 The Model | 19 |
| 2.3 Generalized M-Fluctuation Processes | 20 |
| 2.3.1 Theoretical Fluctuation Processes | 20 |
| 2.3.2 Empirical Fluctuation Processes | 21 |
| 2.3.3 Local Alternatives | 23 |
| 2.3.4 Other Fluctuation Processes | 25 |

| | | |
|----------|--|-----------|
| 2.4 | Generalized M-Fluctuation Tests | 26 |
| 2.4.1 | Choice of the Scores ψ | 26 |
| 2.4.2 | Test Statistics | 28 |
| 2.4.3 | Special Cases | 30 |
| 2.5 | Testing for Parameter Instability in (Generalized) Linear Re- gression Models | 31 |
| 2.5.1 | The General Linear Regression Model | 33 |
| 2.5.2 | The Generalized Linear Regression Model | 36 |
| 2.5.3 | Dependent Data | 39 |
| 2.6 | Applications | 40 |
| 2.6.1 | German M1 Money Demand | 40 |
| 2.6.2 | Illegitimate Births in Großarl | 42 |
| 2.6.3 | Artificial Binary Data | 43 |
| 2.6.4 | Boston Homicide Data | 46 |
| 2.7 | Conclusions | 47 |
| 3 | Implementation of Tests for Structural Change in the R Package | |
| | strucchange | 49 |
| 3.1 | Introduction | 49 |
| 3.2 | The Model | 51 |
| 3.3 | The R System | 52 |
| 3.4 | The Data | 53 |
| 3.5 | Generalized Fluctuation Tests | 57 |
| 3.5.1 | Empirical Fluctuation Processes: Function <code>efp</code> | 57 |
| 3.5.2 | Boundaries and Plotting | 63 |

| | | |
|----------|---|------------|
| 3.5.3 | Significance Testing with Empirical Fluctuation Processes | 68 |
| 3.6 | <i>F</i> Tests | 70 |
| 3.6.1 | <i>F</i> Statistics: Function <i>Fstats</i> | 71 |
| 3.6.2 | Boundaries and Plotting | 72 |
| 3.6.3 | Significance Testing with <i>F</i> Statistics | 73 |
| 3.7 | Monitoring with the Generalized Fluctuation Test | 74 |
| 3.8 | Conclusions | 79 |
| 4 | Testing and Dating of Structural Changes in Practice | 81 |
| 4.1 | Introduction | 81 |
| 4.2 | Model and Methods | 83 |
| 4.2.1 | The Segmented Regression Model | 83 |
| 4.2.2 | Testing Multiple Structural Changes | 84 |
| 4.2.3 | Dating Multiple Structural Changes | 84 |
| 4.3 | Implementation in <i>strucchange</i> | 86 |
| 4.4 | The Nile Data | 88 |
| 4.5 | The Seatbelt Data | 93 |
| 4.6 | The Oil Price Data | 97 |
| 5 | Summary | 101 |
| A | Implementation Details for <i>p</i> Values | 105 |
| B | <i>strucchange</i> Reference Manual | 107 |
| | Bibliography | 172 |

List of Figures

174

Abstract

An approach to testing for structural change in parametric models with a special emphasis on (generalized) linear regression models is introduced: First, a generalized framework for testing for parameter instability based on the partial sums of M-estimation equations is presented which allows for testing the stability of parameter coefficients derived from various estimation techniques like ordinary least squares, maximum likelihood and M-estimation. The core idea of the procedure is to capture the fluctuation in the estimating equations and visualize these and assess their significance. Second, these generalized M-fluctuation tests and virtually all other standard tests for structural change in linear regression models are implemented in a unified framework in the R package `strucchange` reflecting the common features of the testing procedures. Third, the usefulness of the proposed procedures as well as a methodology for recovering the breakpoints of a multiple structural change model are illustrated using several "real world" data sets.

Zusammenfassung

Diese Arbeit stellt eine Methode zum Testen auf Strukturbruch bzw. Strukturveränderung in parametrischen Modellen vor unter besonderer Berücksichtigung von (verallgemeinerten) linearen Regresionsmodellen: Als erstes wird eine allgemeine Methode zur Konstruktion von Tests auf Parameterinstabilität vorgestellt, die auf Partialsummenprozessen von Schätzgleichungen für M-Schätzer basiert. Diese ermöglichen es, die Stabilität von Parameterschätzungen zu testen, die durch verschiedene Schätztechniken wie etwa gewöhnliche Kleinste-Quadrate-, Maximum-Likelihood- oder M-Schätzung ermittelt wurden. Die Grundidee des Verfahrens ist es, Prozesse zu berechnen, die die Fluktuation in den zu Grunde liegenden Schätzgleichungen einfängt, diese zu visualisieren und auf signifikante exzessive Fluktuation zu testen, die wiederum auf Parameterinstabilität schließen ließe.

Als zweites werden diese verallgemeinerten M-Fluktuationstests zusammen mit nahezu allen anderen Standard-Testverfahren auf Strukturbruch in linearen Regresionsmodellen im R-Paket `strucchange` implementiert, das vereinheitlichte Werkzeuge zur Berechnung, Visualisierung und Bestimmung der Signifikanz der verschiedenen Strukturbruchtests zur Verfügung stellt.

Als drittes, wird die Nützlichkeit der behandelten Testverfahren in praxis-

nahen Anwendungen illustriert, wo neben den Tests auch Funktionalität zur Schätzung der Anzahl und Zeitpunkte multipler Brüche in linearen Regressionsmodellen vorgestellt wird.

Chapter 1

Introduction

This monograph is concerned with testing for structural change in parametric models with a special emphasis on linear regression models: First, the theory for a general class of tests for structural change is established. Second, it is discussed how these and other tests for structural change are implemented in a unified framework reflecting the common properties of the tests in the R package `strucchange`. Third, the usefulness of the proposed procedures in connection with the interface of the implementation is illustrated using applications to several data sets in which the structural changes can be attributed to identifiable historical events.

Before discussing some features and properties of tests for structural change more generally, it should be briefly clarified why we need such tests and implementations and applications for them.

What is a structural change and why should we want to test for it? Structural change is of central interest in many fields of research and data analysis: to learn if, when and how the structure of the data generating mechanism underlying a set of observations changes. Usually, it is known with

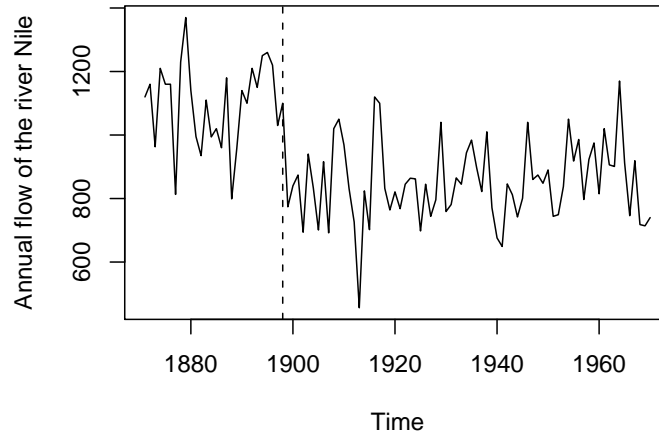


Figure 1.1: Annual flows of the river Nile before and after opening of the first Aswan dam

respect to which quantity the structural change might occur, e.g., over time or with the increase of a certain risk factor. One of the simplest examples for such a structural change is a time series whose mean changes at a single breakpoint. Such a time series is depicted in Figure 1.1 giving the annual discharge of the river Nile at Aswan in 10^8 m^3 before and after the opening of the first Aswan dam in 1898. The plot suggests that the annual flows vary around a constant mean flow in each segment—before 1898 and afterwards—but that there is one break at which the mean flow drops due to the opening of the Aswan dam (in Chapter 4 it is shown that this is indeed a reasonable model for the data). But to assess whether there is evidence for such a structural change or not, a statistical test is needed: given a model (in the example: constant mean flow) it is tested whether the data support the hypothesis that there is a stable structure against the

alternative that it changes over time.

Why do we need an implementation of such tests? The obvious first answer to this question is, of course, that this is the easiest way to enable potential users to adopt some theoretical results in the analysis of data. Another important issue is the reproducibility of research results: if numerical results are given in a scientific article not only the proofs for the theory underlying it should be made available but also the software which produced these results (Leisch 2002). But we would like to stress another point about the implementation of statistical methodology which is often neglected: to a considerable extent software affects how statisticians think about a certain methodology and how they use it in the analysis of data. Therefore, a unified but flexible implementation that reflects the common features of a set of techniques can help practitioners and researchers to understand the underlying theory better and more quickly and to apply the methodology properly and in a more efficient manner.

What are the benefits of applications of the tests? This question is much easier to answer than the previous two. Applications show that the testing procedures lead to valuable and interpretable results in the analysis of “real” data. Furthermore, they convey a certain “feeling” for the way the techniques can be used in different situations.

1.1 Tests for Structural Change

Starting from the recursive CUSUM test of [Brown, Durbin, and Evans \(1975\)](#) a large variety of tests for structural change has been suggested in both the econometrics and the statistics literature many of which can be broadly placed into two different classes: generalized fluctuation tests

(Kuan and Hornik 1995) that do not assume a particular pattern of deviation from the hypothesis of parameter constancy and F tests (Andrews 1993; Andrews and Ploberger 1994) that are built for a single shift alternative (of unknown timing).

For a recent review of F tests for structural change and related techniques see Hansen (2001).

The generalized fluctuation tests fit a parametric model to the data via ordinary least squares (OLS)—or equivalently via maximum likelihood (ML) using a normal approximation—and derive a process which captures the fluctuation of the recursive or OLS residuals (Brown et al. 1975; Ploberger and Krämer 1992; Chu, Hornik, and Kuan 1995a) or the recursive or rolling/moving estimates (Ploberger, Krämer, and Kontrus 1989; Chu, Hornik, and Kuan 1995b) and reject if this fluctuation is improbably large. In their seminal paper Brown et al. (1975) point out that this framework ...

“...includes formal significance tests but its philosophy is basically that of data analysis as expounded by Tukey (1962). Essentially, the techniques are designed to *bring out departures from constancy in a graphic way* instead of parametrizing particular types of departure in advance and then developing formal significance tests intended to have high power against these particular alternatives. From this point of view the significance tests suggested should be regarded as yardsticks for the interpretation of data rather than leading to hard and fast decisions.” (Brown et al. 1975, pp. 149–150)

This emphasizes two points we will discuss in more detail below: first,

visualization of fluctuation tests is important, and second, the tests are not only significance tests but also explorative tools.

Instead of capturing the fluctuation in residuals or parameter estimates empirical fluctuation processes can also be based on OLS first order conditions or ML scores respectively (Nyblom 1989; Hansen 1992; Hjort and Koning 2002). Although they have not been discussed in the generalized fluctuation test framework (Kuan and Hornik 1995) we show below that they can be seen as an extension of that framework which makes the class of empirical fluctuation processes richer. Furthermore, we show how these ideas can be used in more general situations and employing different estimation techniques (not only OLS and ML) under mild assumptions. The resulting class of tests for structural change is based on processes which capture the fluctuation in M-estimating scores.

1.2 Diagnostic Checking vs. Data Exploration

In parametric models structural change is typically described by parameter instability. If this instability is ignored, parameter estimates are generally not meaningful, inference is severely biased and predictions lose accuracy. As Breiman (2001) criticizes, parametric data models in general and the linear regression model in particular are used too often without checking the fit of the model to the data: “The linear regression model led to many erroneous conclusions that appeared in journal articles waving the 5% significance level without knowing whether the model fit the data” (Breiman 2001, p. 203). Furthermore, he argues that it is important to assess the goodness-of-fit of data models not only using omnibus tests but tests designed for a certain direction of the alternative. To avoid using non-

sense models, a large literature on diagnostic tests and misspecification tests that test the assumptions of the linear regression model emerged in particular in the econometrics community. [Krämer and Sonnberger \(1986\)](#) discuss a collection of tests for diagnostic checking in linear regression relationships which are designed for testing the null hypothesis that a standard linear regression model produces an adequate fit for the data against the alternative that one specific assumption is violated, e.g., misspecification, omitted regressors, functional form, heteroskedasticity or autocorrelation of the errors, or structural change.

But as [Brown et al. \(1975\)](#) emphasize for the recursive CUSUM test, none of these tests should only be used as pure significance tests but also as explorative tools which can be used to learn about the structure in data. Rejection of the null hypothesis can not necessarily be regarded as evidence that the particular alternative for which the test is built is true, e.g., structural changes can often also be interpreted as omitted regressors and vice versa, or heteroskedasticity and structural change might be confounded etc. Especially in a structural change framework, it is usually not only of interest to find some model that fits the data (with meaningful parameter estimates, inference and prediction methods) but also to be able to identify and interpret the structural changes which are of high interest for practitioners and researchers. One example for such a situation is the Nile data described above; further examples are discussed throughout the rest of this monograph. Therefore, as [Zeileis and Hothorn \(2002\)](#) point out describing the implementation of a collection of diagnostic tests from [Krämer and Sonnberger \(1986\)](#) in the R package `lmtest`, it is always helpful to use diagnostic tests as explorative tools together with diagnostic plots that help to visualize and understand the structure in the data.

1.3 Visualization and Significance Testing

Although [Brown et al. \(1975\)](#) stressed that the recursive CUSUM test should bring out departures from parameter instability in a graphic way the visualization of structural change tests was often neglected in the literature although a synthesis of visualization and significance testing offers many advantages. For example, [Meyer, Zeileis, and Hornik \(2003\)](#) discuss how goodness-of-fit testing and visualization of residuals in log-linear models can be combined such that a plot reports unusual observations if and only if the corresponding tests rejects the null hypothesis at some specified level. A similar approach is used below for structural change tests.

More generally, [Cleveland \(1993\)](#) describes the interplay of visualization and probabilistic inference:

“Probabilistic inference is the classical paradigm for data analysis in science and technology. [...] Visualization—with its two components, graphing and fitting—is a different paradigm for learning from data. It stresses a penetrating look at the structure of the data. What is learned from the look is guided by knowledge of the subject under study. Sometimes visualization can fully replace the need for probabilistic inference. We visualize the data effectively and suddenly, there is what Joseph Berkson called *interocular traumatic impact*: a conclusion that hits us between the eyes. In other cases, visualization is not enough and probabilistic inference is needed to help calibrate the uncertainty of a less certain issue. When this is so, visualization has yet another role to play—checking assumptions.”
([Cleveland 1993](#), pp. 12–14)

Visualization of structural change tests as introduced in this monograph uses this methodology—fitting of a model to the data and graphing of the fluctuation in residuals, estimates or M-scores—and combines all three purposes: formal significance testing, visualization of the structure of the data and checking the assumption of parameter stability in the underlying model. Additionally, the examples discussed throughout this monograph illustrate what is called “interocular traumatic impact” above because the plots of fluctuation processes not only visualize the result of the corresponding significance test but also convey rather precise information about the type and number of structural changes.

1.4 Segmentation of Regression Models

Given that there is significant evidence for structural change in some (regression) model, a natural approach is to estimate the breakpoints from the data and fit a model to each segment defined by these breakpoints. In several statistical disciplines and communities, methods have been developed that address this problem from different viewpoints. Although the terminology is usually quite different, the corresponding techniques are often strongly related.

In statistical quality control and sequential testing, the detection of such abrupt changes in a sequence of observations has always been an important topic (see e.g., [Siegmund 1985](#); [Basseville and Benveniste 1986](#); [Basseville and Nikiforov 1993](#)) which aims at finding out when the underlying distribution in this sequence changes as soon as possible and adjust the model correspondingly. In medical research, maximally selected rank statistics are frequently used to estimate cutpoints in some prognostic fac-

tor that splits the sample into two parts and thus serves as a simple binary segmentation rule (Lausen and Schumacher 1992; Hothorn and Lausen 2003). Similar techniques can also be applied recursively with respect to more than one explanatory variable which leads to recursive partitioning and classification and regression trees (Breiman, Friedman, Olshen, and Stone 1984) where usually a constant is fitted to the data in each segment. Instead of segmenting with respect to several variables and fitting simple (constant) models to each segment, another approach is to find change-points with respect to only one variable and fit a more sophisticated model in each segment, e.g., a (generalized) linear regression model (Stasinopoulos and Rigby 1992; Pastor-Barriuso, Guallar, and Coresh 2003). To determine the breakpoints in such a model algorithms are needed which minimize the likelihood (Hawkins 2001) or residual sum of squares (Bai and Perron 1998, 2003). This is the natural approach in a structural change framework where breaks in a regression model fitted by ML or OLS should be discovered with respect to a unique ordering of the data. If time is the ordering variable this procedure is often called dating of structural changes. We adopt this approach and implement and apply the algorithm of Bai and Perron (2003) in such a way that it reflects the common features of the procedures mentioned above: given evidence for structural breaks, find a segmentation with respect to one ordering variable and then fit a segmented regression model.

1.5 Overview

This monograph presents a theoretically and computationally sound and flexible framework that reflects all the features described above: First,

it extends the generalized fluctuation test framework ([Kuan and Hornik 1995](#)) by introducing fluctuation processes based on M-scores. Second, it describes how (M-)fluctuation tests on the one hand and F tests ([Andrews 1993](#); [Andrews and Ploberger 1994](#)) on the other are implemented in a unified way in the R package `strucchange`. Third, the tests are applied to data sets with a special focus to multiple change models and the recovering of the number and location of the breakpoints. More precisely, this monograph is organized as follows: Chapter 2 introduces a general class of fluctuation tests for parameter instability which is based on partial sums of M-estimation equations and contains many well-known tests from the generalized fluctuation test framework ([Kuan and Hornik 1995](#)) as special cases. Functional central limit theorems which govern the asymptotic behaviour of the corresponding fluctuation processes under the hypothesis and local alternatives are derived under mild assumptions. A unified approach is outlined to the construction of test statistics and to strategies for combining traditional significance testing with visualization methods for detecting the timing of a potential shift and which parameter is affected by it. Finally, a few applications of the new fluctuation tests are given. The chapter essentially presents the results of [Zeileis and Hornik \(2003\)](#).

Chapter 3 describes the implementation of a large collection of tests for structural change encompassing virtually all commonly used structural change tests in the R package `strucchange`. Usually, standard econometrics and statistics software packages just implement a few single tests for structural change which do not reflect the common features of the tests and seldom offer automatic visualization of the results. Therefore, the `strucchange` package implements classes of tests (rather than single tests) and unifies the approaches to testing for structural change with tests from

the generalized (M-)fluctuation test framework (Kuan and Hornik 1995; Zeileis and Hornik 2003) and the F test framework (Andrews 1993; Andrews and Ploberger 1994): first a fluctuation process or sequence of F statistics is computed which then can be plotted together with its boundaries and finally a significance test can be carried out, including the computation of an (approximate) asymptotic p value. This connects traditional significance testing with visualization techniques that allow for better understanding of the structure of the data and the decision of the corresponding significance test, like motivated above. Furthermore, it is described how incoming data can be monitored online. The chapter mainly gives the results of Zeileis, Leisch, Hornik, and Kleiber (2002).

Chapter 4 applies the introduced methodology for testing for structural change to “real world” data with a special emphasis on multiple structural changes and dating of structural changes in a linear regression setup. In addition to the various testing strategies, a dynamic programming algorithm for the estimation of the breakpoints is discussed and implemented in the `strucchange` package. Using historical data on Nile river discharges, road casualties in Great Britain and oil prices in Germany it is shown that statistically detected changes in the mean of a time series as well as in the coefficients of a linear regression coincide with identifiable historical, political or economic events which might have caused these breaks. The results are essentially those of Zeileis, Kleiber, Krämer, and Hornik (2003).

Chapter 2

Generalized M-Fluctuation Tests for Parameter Instability

2.1 Introduction

The generalized fluctuation test framework ([Kuan and Hornik 1995](#)) is one of the important classes of structural change tests as discussed above. Its main idea is to fit a parametric model to the data via OLS—or equivalently via maximum likelihood (ML) using a normal approximation—and derive a process which captures the fluctuation in the residuals or parameter estimates and reject if this fluctuation is improbably large. Using similar principles, tests for parameter instability have been suggested that capture fluctuations in OLS first order conditions or ML scores respectively ([Nyblom 1989](#); [Hansen 1992](#); [Hjort and Koning 2002](#)). Although these have not been discussed in the generalized fluctuation test framework ([Kuan and Hornik 1995](#)) we show below that they can be seen as an extension of that framework which makes the class of empirical fluctuation

tuation processes richer. In addition, we further unify the fluctuation test framework by showing how under mild assumptions these ideas can be used in more general situations and employing different estimation techniques (not only OLS and ML). The resulting class of tests for parameter instability which is based on M-estimation scores contains many of the tests mentioned above as special cases and unifies the approaches to the construction of test statistics. Strategies are outlined for combining traditional significance testing with visualization methods for detecting the timing of a potential structural change and which parameter is affected by it.

All these techniques have in common that it is known with respect to which quantity the instability might occur: e.g., in time series regression it is natural to ask whether the relationship between dependent and explanatory variables changes over time. In clinical studies often change-point problems arise where a regression relationship changes with respect to the size of one risk factor etc. The situations described have in common that the observations have some unique ordering with respect to which a structural change occurs but that the (potential) changepoint is unknown.

This chapter is organized as follows: in Section 2.2 we introduce the parametric model and formulate the null hypothesis before we derive functional central limit theorems for partial sum processes of M-scores under the hypothesis of parameter stability and under local alternatives in Section 2.3. The construction of the generalized M-fluctuation tests—from the choice of the estimation technique to the test statistic that captures the fluctuation in the M-score processes—is described in Section 2.4. Section 2.5 discusses tests based on ML-scores in (generalized) linear models and the usefulness of the proposed tests is illustrated in Section 2.6 based on data

for German M1 money demand, historical demographic time series of illegitimate births in the village of Großarl, Austria, and youth homicides in Boston, USA, in a policy intervention framework.

2.2 The Model

We assume n independent observations

$$Y_i \sim F(\theta_i) \quad (i = 1, \dots, n). \quad (2.1)$$

distributed according to some distribution F with k -dimensional parameter θ_i . We also assume that the observations are uniquely ordered by some external variable, usually time. The Y_i can possibly be vector valued, extensions to a regression situation where $Y_i = (y_i, x_i)^\top$ and the x_i are some additional covariates are presented later.

We are interested in testing the hypothesis

$$H_0: \theta_i = \theta_0 \quad (i = 1, \dots, n) \quad (2.2)$$

against the alternative that (at least one component of) θ_i varies over “time”. For this alternative to be sensible the ordering assumption is necessary: If a parameter instability with a single breakpoint, say, occurs with respect to a certain ordering of the variables this single breakpoint interpretation would be lost by re-ordering in such a way that observations from the two regimes are mixed. The assumption of independence is assumed for convenience and will be weakened later.

2.3 Generalized M-Fluctuation Processes

In this chapter, we suggest a general class of fluctuation processes that can capture instabilities in the parameter θ . In the first two Sections 2.3.1 and 2.3.2 the fluctuation processes are introduced and their behaviour under the null hypothesis (2.2) is derived. The limiting process is first derived for known θ_0 and then for the case where it has to be estimated. In Section 2.3.3, the results are generalized to local alternatives and finally in Section 2.3.4 some further fluctuation processes are introduced.

2.3.1 Theoretical Fluctuation Processes

Consider some suitably smooth k -dimensional score function $\psi(\cdot)$ (see e.g., White 1994), independent of n and i , with

$$E[\psi(Y_i, \theta_i)] = 0 \quad (2.3)$$

and define the following matrices

$$A(\theta) = E[-\psi'(Y, \theta)], \quad (2.4)$$

$$B(\theta) = \text{COV}[\psi(Y, \theta)], \quad (2.5)$$

$$C(\theta) = E[\psi(Y, \theta)u(Y, \theta)^\top] \quad (2.6)$$

where $Y \sim F(\theta_0)$, $\psi'(\cdot)$ is the gradient of $\psi(\cdot)$ with respect to θ , and $u(\cdot, \theta)$ is

$$u(y, \theta) = \frac{\partial \log f(y, \theta)}{\partial \theta}, \quad (2.7)$$

and $f(\cdot, \theta)$ is the probability density function corresponding to F . Hence, u is the gradient of the log likelihood with respect to θ , also called Maximum Likelihood (ML) score. The first two matrices $A(\theta)$ and $B(\theta)$ are standard

in M-estimation, $C(\theta)$ is only needed in Section 2.3.3. Note that given ψ the matrices $A(\theta)$ and $B(\theta)$ but not $C(\theta)$ can be estimated without further knowledge of F or f respectively as the latter depends on the ML score u .

Theorem 1 *For the cumulative score process given by*

$$W_n(t, \theta) = \frac{1}{\sqrt{n}} \sum_{i=1}^{\lfloor nt \rfloor} \psi(Y_i, \theta) \quad (2.8)$$

and under the assumptions stated above and under H_0 the following functional central limit theorem (FCLT) holds:

$$W_n(\cdot, \theta_0) \xrightarrow{d} Z(\cdot),$$

where $Z(\cdot)$ is a Gaussian process with continuous paths, mean function $E[Z(t)] = 0$ and covariance function $\text{COV}[Z(t), Z(s)] = \min(t, s) \cdot B(\theta_0)$.

Proof: The proof follows by direct application of Donsker's theorem (Billingsley 1999).

Corollary 1 *If $B(\theta_0)$ is non-singular, the following FCLT holds for the decorrelated fluctuation process*

$$B(\theta_0)^{-1/2} W_n(\cdot, \theta_0) \xrightarrow{d} W(\cdot),$$

where $W(\cdot)$ is a k -dimensional Wiener process or standard Brownian motion.

2.3.2 Empirical Fluctuation Processes

Usually, in applications the parameter θ_0 is not known but has to be estimated. A suitable estimator can be based on the function $\psi(\cdot)$: the full sample M-estimator $\hat{\theta}_n$ is defined by the equation

$$\sum_{i=1}^n \psi(Y_i, \hat{\theta}_n) = 0. \quad (2.9)$$

Some properties of this M-estimator are well known (see e.g., [Stefanski and Boos 2002](#)). Taylor expansion of

$$S_n(\theta) = \frac{1}{n} \sum_{i=1}^n \psi(Y_i, \theta) \quad (2.10)$$

gives

$$0 = S_n(\hat{\theta}_n) = S_n(\theta_0) + S'_n(\theta_0)(\hat{\theta}_n - \theta_0) + R_n. \quad (2.11)$$

Under suitable regularity conditions

$$\begin{aligned} -S'_n(\theta_0) &= \frac{1}{n} \sum_{i=1}^n -\psi'(Y_i, \theta_0) \xrightarrow{p} A(\theta_0), \\ \sqrt{n}S_n(\theta_0) &\xrightarrow{d} \mathcal{N}(0, B(\theta_0)), \\ \sqrt{n}R_n &\xrightarrow{p} 0. \end{aligned}$$

Therefore, the following holds

$$\sqrt{n}(\hat{\theta}_n - \theta_0) \xrightarrow{d} \mathcal{N}(0, V(\theta_0)), \quad (2.12)$$

where $V(\theta) = A(\theta)^{-1}B(\theta)\{A(\theta)^{-1}\}^\top$. See [Stefanski and Boos \(2002\)](#) or [White \(1994\)](#) for further details; [White \(1994, Theorem 6.10, p. 104\)](#) also gives a set of suitable regularity conditions. Equivalently to (2.12) we can write

$$\sqrt{n}(\hat{\theta}_n - \theta_0) \doteq A(\theta_0)^{-1} \cdot W_n(1, \theta_0), \quad (2.13)$$

where $a_n \doteq b_n$ means that $a_n - b_n$ tends to zero (in probability if a_n or b_n are stochastic).

Theorem 2 *Under H_0 the following FCLT holds for the empirical cumulative score process with M-estimated parameters*

$$W_n(\cdot, \hat{\theta}_n) \xrightarrow{d} Z^0(\cdot),$$

where $Z^0(t) = Z(t) - tZ(1)$.

Proof:

$$\begin{aligned}
W_n(t, \hat{\theta}_n) &\doteq \frac{1}{\sqrt{n}} \sum_{i=1}^{\lfloor nt \rfloor} \psi(Y_i, \theta_0) + \frac{1}{n} \sum_{i=1}^{\lfloor nt \rfloor} \psi'(Y_i, \theta_0) \cdot \sqrt{n}(\hat{\theta}_n - \theta_0) \\
&\doteq W_n(t, \theta_0) - \frac{\lfloor nt \rfloor}{n} A(\theta_0) \cdot A(\theta_0)^{-1} W_n(1, \theta_0) \\
&\xrightarrow{d} Z(t) - t \cdot Z(1).
\end{aligned}$$

Corollary 2 *If $B(\theta_0)$ is non-singular, the following FCLT holds for the decorrelated empirical fluctuation process with M-estimated parameters*

$$\hat{B}_n^{-1/2} W_n(\cdot, \hat{\theta}_n) \xrightarrow{d} W^0(\cdot),$$

where $W^0(\cdot)$ is a standard k -dimensional Brownian bridge with $W^0(t) = W(t) - tW(1)$ and \hat{B}_n some consistent and non-singular covariance matrix estimate, e.g.,

$$\hat{B}_n = \frac{1}{n} \sum_{i=1}^n \psi(Y_i, \hat{\theta}_n) \psi(Y_i, \hat{\theta}_n)^\top. \quad (2.14)$$

In the following, the empirical fluctuation process is also denoted

$$efp(t) = \hat{B}_n^{-1/2} W_n(t, \hat{\theta}). \quad (2.15)$$

2.3.3 Local Alternatives

In parameter instability problems or structural change situations an alternative of interest is the local alternative

$$H_A : \theta_i = \theta_0 + \frac{1}{\sqrt{n}} g\left(\frac{i}{n}\right), \quad (2.16)$$

where $g(\cdot)$ is a function of bounded variation on $[0, 1]$ which describes the pattern of departure from stability of the parameter θ_0 (Kuan and Hornik 1995; Hjort and Koning 2002).

Then Y_i has the probability density function

$$f(y, \theta_i) \doteq f(y, \theta_0) \left\{ 1 + u(y, \theta_0)^\top \frac{1}{\sqrt{n}} g\left(\frac{i}{n}\right) \right\}, \quad (2.17)$$

which can be easily derived from first order Taylor expansion of f .

Therefore, under a local alternative like (2.16) the components of the fluctuation process (2.8) no longer have zero mean in general but

$$E[\psi(Y_i, \theta_0)] \doteq \int \psi(y, \theta_0) f(y, \theta_0) \, dy + \quad (2.18)$$

$$\int \psi(y, \theta_0) u(\theta_0)^\top f(y, \theta_0) \frac{1}{\sqrt{n}} g\left(\frac{i}{n}\right) \, dy \quad (2.19)$$

$$= 0 + \frac{1}{\sqrt{n}} C(\theta_0) g\left(\frac{i}{n}\right). \quad (2.20)$$

In fact, with the same arguments as in Section 2.3.1 and 2.3.2, the whole fluctuation process can be split into one part which is governed by the FCLT from Theorem 1 and a second part which is determined by the function g from (2.16):

$$W_n(\cdot, \theta_0) \xrightarrow{d} Z^A(\cdot), \quad (2.21)$$

where $Z^A(t) = W(t) + C(\theta_0)G(t)$ and $G(\cdot)$ is the antiderivative of g with $G(t) = \int_0^t g(y) \, dy$.

Finally, the following limiting process can be derived for the decorrelated empirical fluctuation process:

$$\hat{B}_n^{-1/2} W_n(t, \hat{\theta}_n) \doteq \hat{B}_n^{-1/2} \{W_n(t, \theta_0) - tW_n(1, \theta_0)\} \quad (2.22)$$

$$\doteq W^0(t) + B(\hat{\theta}_n)^{-1/2} C(\hat{\theta}_n) G^0(t), \quad (2.23)$$

with $G^0(t) = G(t) - tG(1)$, provided $B(\cdot)$ is consistent under H_A .

The results above include the results from Section 2.3.1 and 2.3.2 as special cases because under the null hypothesis of parameter stability (2.2) the function g is identical to zero $g \equiv 0$. But the results also imply that tests

based on the empirical fluctuation processes will be consistent against suitable local alternatives of type (2.16).

2.3.4 Other Fluctuation Processes

Instead of capturing the fluctuation in a cumulative sum of scores, a moving or rolling sum of scores could be used as well. More formally, we also consider processes of type

$$M_n(t, \theta) = \frac{1}{\sqrt{n}} \sum_{i=\lfloor N_n t \rfloor + 1}^{\lfloor N_n t \rfloor + \lfloor nh \rfloor} \psi(Y_i, \theta), \quad (2.24)$$

where $N_n = (n - \lfloor nh \rfloor) / (1 - h)$ and h determines the bandwidth.

Under H_0 this process converges to the increments of a Brownian motion or bridge respectively, depending on whether the value θ_0 is known or has to be estimated. The latter case is stated formally in the following theorem.

Theorem 3 *Under H_0 the following FCLT holds for the empirical moving score fluctuation process:*

$$\hat{B}_n^{-1/2} M_n(\cdot, \hat{\theta}_n) \xrightarrow{d} M^0(\cdot), \quad (2.25)$$

where $M^0(t) = W^0(t + h) - W^0(t)$ is the process of increments of a Brownian bridge.

Proof: The proof follows by application of Lemma A from [Chu et al. \(1995a\)](#) to the results of Section 2.3.2 and 2.3.1.

2.4 Generalized M-Fluctuation Tests

2.4.1 Choice of the Scores ψ

We choose the M-estimation framework for estimation of the parameters θ as it contains many other estimation techniques as special cases by choosing a suitable score function ψ . Other classes of estimators are not strictly special cases but are strongly related to M-estimation and the principles introduced in Section 2.3 can be used to construct fluctuation processes with the same asymptotic properties. A few of these generalizations are outlined in the following.

One of the most common choices for ψ is the partial derivative of some objective function Ψ

$$\psi(y, \theta) = \frac{\partial \Psi(y, \theta)}{\partial \theta}, \quad (2.26)$$

where Ψ could be the residual sum of squares or the Log-Likelihood, yielding the OLS or ML estimators $\hat{\theta}$ respectively (the dependence of $\hat{\theta}$ on the number of observations n is ignored in the following). In both cases the cumulative sums of the first order conditions ψ lead very naturally to fluctuation processes as described in the previous section.

Another estimating approach which is particularly popular in econometrics is to use a Quasi-Maximum Likelihood (White 1994) in a misspecification context. This again is similar to M-estimation in robust statistics (Huber 1964, 1972) which also accounts for violation of some of the standard model assumptions. Huber (1964) suggests the function

$$\psi_H(y, \theta) = \min(c, \max(y - \theta, -c)). \quad (2.27)$$

with some constant c for robust estimation of the mean of a symmetric distribution. Note that this function is not smooth but it is almost every-

where differentiable (except in $\pm c$) and in the definition of the matrix $A(\theta)$ in (2.4) integration and differentiation can be interchanged. Hence, almost identical results for estimation and construction of fluctuation processes can be derived using functions like Huber's ψ from (2.27).

Another approach is to not fully specify a model via its likelihood but via some estimating equations (Godambe 1960, 1985) which are satisfied by the true model. Similarly, some moment or orthogonality conditions can be exploited to derive estimating functions which again yield parameter estimates. This approach is used in estimation techniques like instrumental variables in linear models (IV, Sargan 1958), the generalized method of moments (GMM, Hansen 1982) for the estimation of economic models or the generalized estimating equations (GEE, Liang and Zeger 1986) for models for longitudinal or time-series data in biostatistics. Further discussion of related estimation approaches can be found in Bera and Biliias (2002). Usually, these are regression models which are not yet covered by the methodology introduced above. However, with some modifications as described in Section 2.5 fluctuation processes with rather similar properties can be derived.

All these methods have in common that the estimation of θ is based on some score or estimating function ψ or a moment or orthogonality condition similar to (2.9) whose partial sums yield fluctuation processes satisfying some FCLT which again can be used to construct tests for parameter instability. The latter step will be described in detail in the following section.

2.4.2 Test Statistics

We derived the empirical fluctuation processes because they can capture departures from the null hypothesis (2.2) of parameter stability. Therefore, visual inspection alone conveys information about whether H_0 is violated or not. But this alone is, of course, not enough and we want to derive tests based on empirical fluctuation processes. One common strategy for this is to consider some scalar functional λ that can be applied to the fluctuation processes.

Given a finite sample as in model (2.1) an empirical fluctuation process is an $n \times k$ matrix $(efp_j(i/n))_{i,j}$ with $i = 1, \dots, n$ and $j = 1, \dots, k$ that converges to a k -dimensional limiting process which is continuous in time. To aggregate this empirical process to a scalar test statistic several suitable functionals of the form

$$\lambda \left(efp_j \left(\frac{i}{n} \right) \right). \quad (2.28)$$

are conceivable. The limiting distribution for these test statistics can be determined fairly easily, it is just the corresponding (asymptotic) functional applied to the limiting process. Although closed form results for certain functionals of Brownian bridges exist, the critical values are typically best derived by simulation so there are no constraints for the choice of λ .

λ can usually be split into two components: λ_{time} which aggregates over time and λ_{comp} which aggregates over the components of ψ . Common choices for λ_{time} are the absolute maximum, the mean or the range (Krämer and Schotman 1992; Kuan and Hornik 1995; Hjort and Koning 2002; Zeileis et al. 2002). Typical functionals λ_{comp} include the maximum norm (or L_∞ norm, denoted as $\|\cdot\|_\infty$) or the squared Euclidian norm (or L_2 norm, denoted as $\|\cdot\|_2^2$), see Hjort and Koning (2002) for more examples.

As the decorrelated processes are asymptotically independent it seems to be more intuitive to first aggregate over time and then have k independent univariate test statistics, each associated with one component of the process which can usually be matched with one component of the parameter vector θ . If the overall hypothesis is rejected the component(s) of θ which caused the instability can then be identified.

On the other hand, when there is evidence for a structural change a very natural question is *when* it occurred. To focus on this question it is obviously better to first aggregate over j and then inspect the resulting univariate process for excessive fluctuation which can be also done visually, e.g. by checking whether this process crosses some boundary $b(t) = c \cdot d(t)$. In this case c determines the significance level and $d(\cdot)$ the shape of the boundary and the resulting test statistic is

$$\max_{i=1,\dots,n} \left| \frac{\lambda_{\text{comp}} \left(\text{efp}_j(i/n) \right)}{d(i/n)} \right|, \quad (2.29)$$

i.e., a weighted maximum of the absolute values of the process aggregated by λ_{comp} . Natural choices are to weigh all observations equally, i.e., $d(t) = 1$, or by the (asymptotic) standard deviation of the fluctuation process, i.e., $d(t) = \sqrt{t(1-t)}$ for the cumulative score process. But other boundaries are also conceivable, see [Zeileis \(2004\)](#) or [Zeileis, Leisch, Kleiber, and Hornik \(2004\)](#) for a more detailed discussion.

The only class of test statistics which allows for both identification of the component j as well as the timing i/n of a potential structural instability is when the maximum is used for aggregating over both time and components, i.e.,

$$\max_{i=1,\dots,n} \max_{j=1,\dots,k} \left| \frac{\text{efp}_j(i/n)}{d(i/n)} \right| \quad (2.30)$$

where the $efp_j(i/n)$ which cross some absolute critical value c can be regarded as violating the hypothesis of stability (Mazanec and Strasser 2000).

2.4.3 Special Cases

The rich class of generalized M-fluctuation tests introduced in this chapter contains various tests for parameter instability or structural change known from the statistics and econometrics literature.

ML scores: Most importantly, the generalized M-fluctuation tests contain the tests of Hjort and Koning (2002) who develop a general class of fluctuation processes based on the ML scores from (2.7). These yield the ML estimate of θ_0 and $A(\theta_0) = B(\theta_0) = C(\theta_0) = I(\theta_0)$ is the usual Fisher information matrix. Hjort and Koning (2002) illustrate how to construct three types of tests which are all included in the more general framework above. In particular they construct a Cramér-von Mises type test which is the average of the Euclidian norm of the fluctuation process at time t . However, they do not point out that this is the test of Nyblom (1989) which Hansen (1992) generalized to linear regression models. Nyblom (1989) showed that this test is locally most powerful against the alternative that the parameters follow a random walk.

Changes in the mean: In the case that the Y_i are (not necessarily normally) distributed with mean μ_0 and variance σ^2 various tests for the constancy of the mean can be shown to be special cases of the approach presented above: if μ_0 is estimated by means of OLS the OLS-based CUSUM test (Ploberger and Krämer 1992) and the recursive estimates test (Ploberger et al. 1989) are both equivalent to the natural test resulting from the ideas described above: derive the partial sum process of the scores from Corol-

lary 2 and reject if the maximum absolute value of the process—which is just the scaled cumulative sum of the OLS residuals—is too large. With boundaries proportional to the standard deviation the resulting test is the alternative OLS-based CUSUM test of Zeileis (2004) and with moving instead of cumulative sums as in (2.25) the resulting process is equivalent to the OLS-based MOSUM process (Chu et al. 1995a) and the moving estimates process (Chu et al. 1995b). Krämer and Schotman (1992) and Kuan and Hornik (1995) also consider tests based on the range instead of maximum absolute value for the same processes. If robust M-estimation instead of OLS is used for estimating μ_0 tests like the (non-recursive) robust CUSUM tests of Sen (1984) or Sibbertsen (2000) can be constructed.

The connection between the generalized M-fluctuation tests and the OLS-based CUSUM test and the Nyblom-Hansen test (Cramér-von Mises type test) respectively in a linear regression framework will be described in more detail in the following section.

2.5 Testing for Parameter Instability in (Generalized) Linear Regression Models

The general framework for constructing fluctuation processes and tests based on M-scores presented in the first sections of this chapter is already extremely useful for testing the constancy of model parameters over time; but to explain the generality of the approach without too much technical overhead two assumptions which do not really restrict the generality of the results have been made for convenience. These will be weakened in this Section. First, the observations were assumed to be independent—

an assumption which is likely to be violated in time series applications but which can usually be overcome easily as shown at the end of this section. Second, the probability density function $f(y_i, \theta_i)$ was assumed to describe the full distribution of the Y_i —but if the observations can be split into $Y_i = (y_i, x_i)^\top$ with a response or dependent variable y_i and additional regressors or covariates x_i the usual approach is to model the conditional distribution $f(y_i | x_i, \theta_i)$ given the x_i . The common assumption is that the x_i form a weakly dependent process without deterministic or stochastic trends, see [Andrews \(1993\)](#) for technical details or also [Hansen \(1992\)](#) or [Hjort and Koning \(2002\)](#). Under such suitable assumptions the same asymptotic distribution can be derived for the processes based on estimates of the regression coefficients which can be obtained by various procedures as discussed in [Section 2.4.1](#).

To make the dependence on the covariates obvious the score function ψ from [\(2.3\)](#) is now written as

$$\psi(Y_i, \theta_i) = \psi(y_i, x_i, \theta_i). \quad (2.31)$$

Of course, it is still required to have zero expectation (with respect to $f(y_i | x_i, \theta_i)$) which is not difficult to obtain, more crucial is the assumption that the variances stabilize:

$$\frac{1}{n} \sum_{i=1}^n \text{COV}[\psi(y_i, x_i, \theta_0)] = J_n \xrightarrow{P} J, \quad (2.32)$$

where the matrix J in a regression context corresponds to $B(\theta)$ from [\(2.5\)](#) in the no-covariate context. This follows for example from the weak dependence assumption stated above.

It is easy to show that functional central limit theorems similar to [Theorem 1](#) and [2](#) hold for the resulting theoretical and empirical fluctuation

processes based on $\psi(y_i, x_i, \theta_i)$, in particular the limiting processes are the same with J instead of $B(\theta_0)$. In the following, two important classes of regression models—the general linear model (LM) and the generalized linear model (GLM)—are treated in more detail.

2.5.1 The General Linear Regression Model

Consider the general linear regression model

$$y_i = x_i^\top \beta + u_i \quad (i = 1, \dots, n), \quad (2.33)$$

where the disturbances have zero mean and common variance σ^2 . The precise formulation of these assumptions is not important as long as they imply the same FCLT—for different sets of assumptions see, e.g., [Hansen \(1992\)](#), [Andrews \(1993\)](#), [Ploberger and Krämer \(1992\)](#) or [Bai \(1997a\)](#). To simplify notation and to emphasize common properties of the scores in the LM and the GLM the mean of the y_i is sometimes denoted by $\mu_i = x_i^\top \beta$.

The model parameters $\theta = (\beta, \sigma^2)^\top$ are usually estimated by OLS or ML (based on a normal model) which is both equivalent to using the following scores (summands of the first order conditions in an OLS framework):

$$\psi(y_i, x_i, \theta) = \left(\psi_\beta(y_i, x_i, \beta), \psi_{\sigma^2}(y_i, x_i, \beta, \sigma^2) \right)^\top, \quad (2.34)$$

$$\psi_\beta(y_i, x_i, \beta) = x_i(y_i - x_i^\top \beta) = x_i(y_i - \mu_i), \quad (2.35)$$

$$\psi_{\sigma^2}(y_i, x_i, \beta, \sigma^2) = (y_i - x_i^\top \beta)^2 - \sigma^2. \quad (2.36)$$

These give the usual estimates $\hat{\beta}$ and $\hat{\sigma}^2$. In a normal model the two estimates are independent and thus the covariance matrix J_n corresponding to $\psi(\cdot)$ from (2.34) is block diagonal

$$J_n = \sigma^2 \begin{pmatrix} \frac{1}{n} \sum_{i=1}^n x_i x_i^\top & 0 \\ 0 & 2\sigma^2 \end{pmatrix}.$$

Therefore, independent test statistics for the constancy of β and σ respectively can be computed and the stability of the parameters can be assessed independently.

Three test statistics will be derived in the following: the Nyblom-Hansen test, the double max test and the OLS-based CUSUM test.

Nyblom-Hansen test: We follow the approach of Hansen (1992) and test both parameters $\theta = (\beta, \sigma^2)^\top$ simultaneously. In his equation (9) he gives the formula for the test statistic L_C based on the following empirical fluctuation process and covariance estimate

$$W_n(t, \hat{\theta}) = \frac{1}{\sqrt{n}} \sum_{i=1}^{\lfloor nt \rfloor} \psi(y_i, x_i, \hat{\theta}), \quad (2.37)$$

$$\hat{J} = \frac{1}{n} \sum_{i=1}^n \psi(y_i, x_i, \hat{\theta}) \psi(y_i, x_i, \hat{\theta})^\top. \quad (2.38)$$

The test statistic L_C is then given as

$$\begin{aligned} L_C &= \frac{1}{n} \sum_{i=1}^n W_n\left(\frac{i}{n}, \hat{\theta}\right)^\top \hat{J}^{-1} W_n\left(\frac{i}{n}, \hat{\theta}\right) \\ &= \frac{1}{n} \sum_{i=1}^n \text{efp}\left(\frac{i}{n}\right)^\top \text{efp}\left(\frac{i}{n}\right) \\ &= \frac{1}{n} \sum_{i=1}^n \left\| \text{efp}\left(\frac{i}{n}\right) \right\|_2^2, \end{aligned}$$

which can be interpreted easily as described in Section 2.4.2. The asymptotic distribution is $\int_0^1 \|W^0\|_2^2$, where W^0 is a k -dimensional Brownian bridge. Nyblom (1989) first suggested this test in a structural change context and showed that it is locally most powerful for the alternative that the parameters follow a random walk. Without relating to Nyblom's earlier work, Hjort and Koning (2002) refer to it as a Cramér-von Mises type test, but they also point out that visual inspection of the process $\text{efp}(t)$ might convey information about the timing of a potential structural change. Al-

though enhancing classical significance testing by visual means is a good idea, this approach has the problem that what is tested and what is visualized differ. From the transformation above it becomes clear that it is much more natural to use a plot of $\|efp(t)\|_2^2$ with two horizontal lines, one for the empirical mean and one for the critical value which visualizes both the significance test and excessive fluctuation (i.e., information about the timing of the shift). An example for this fluctuation process and its visualization is given in Section 2.6.1.

Double max test: As mentioned in Section 2.4.2 the only test statistic which allows the identification of both the timing of a structural change and the component of the parameter vector θ which has changed is

$$\max_{j=1,\dots,k} \max_{0 \leq t \leq 1} |efp_j(t)|, \quad (2.39)$$

which is the same functional for measuring excessive fluctuation as in the recursive estimates test (Ploberger et al. 1989). The limiting distribution is $\max_{j=1,\dots,k} \|W_j^0(t)\|_\infty$. Again, this test can also be performed graphically by plotting each individual process with a horizontal boundary for \pm the critical value.

OLS-based CUSUM test: As already indicated in the previous section, the OLS-based CUSUM test for a change in the mean (without covariates) can be shown to be a special case of the the generalized M-fluctuation test framework. In the regression setup this means that there is one constant regressor $x_i \equiv 1$ and the variance σ^2 is treated as a nuisance parameter. If we consider the OLS-based CUSUM test for a setup with covariates another interesting interpretation emerges from the M-fluctuation view. If an intercept is included in the regression, the OLS-based CUSUM process is equivalent to the first component of the non-decorrelated processes

$W_n(t, \hat{\theta})$ from (2.37) standardized by $\hat{\sigma}^2$ which is element (1, 1) of the estimated covariance matrix \hat{J} . Thus, whereas the first component of the decorrelated fluctuation process $\hat{J}^{-1/2}W_n(t, \hat{\theta})$ captures instabilities of the intercept the first component of the non-decorrelated fluctuation process $W_n(t, \hat{\theta})$ captures instabilities in the (expected) mean $E[x]^\top \beta$. The latter point is proved formally in Ploberger and Krämer (1992). This result implies that shifts orthogonal to the mean regressor $E[x]$ cannot be detected using fluctuation processes based on residuals as such shifts can be interpreted to change the variance rather than the (expected) mean. If the mean L_2 norm rather than the L_∞ norm is used (as in the Nyblom-Hansen test) to measure excessive fluctuation in the OLS-based CUSUM process the resulting test is also trend-resistant, i.e., can deal with trending regressors (Ploberger and Krämer 1996).

2.5.2 The Generalized Linear Regression Model

Now consider the generalized linear model (GLM) like in McCullagh and Nelder (1989). To fix notation, y_i is a response variable distributed independently according to a distribution $F(\theta, \phi)$ where θ is the canonical parameter and ϕ is the dispersion parameter common to all y_i . The probability density has the form

$$f(y_i | \theta, \phi) = \exp \left\{ \frac{y_i \theta - q(\theta)}{w(\phi)} + p(y_i, \phi) \right\}, \quad (2.40)$$

for some known functions $p(\cdot)$, $q(\cdot)$ and $w(\cdot)$, so that $E[y_i] = \mu_i = q'(\theta)$ and $\text{VAR}[y_i] = w(\phi)q''(\theta) = w(\phi)V(\mu_i)$.

The following relationship is assumed for covariates and responses:

$$\mu_i = h(\eta_i) = h(x_i^\top \beta_i) \quad (i = 1, \dots, n), \quad (2.41)$$

where $h^{-1}(\cdot)$ is a known link function, β is again the vector of regression coefficients and η_i is the linear predictor.

The regression coefficients β are usually estimated by ML and ϕ is treated as a nuisance parameter (or is known anyway). The resulting score function for β is

$$\psi(y_i, x_i, \beta) = x_i h'(x_i^\top \beta) V(\mu_i)^{-1} (y_i - \mu_i), \quad (2.42)$$

where $h'(\cdot)$ is the derivative of the inverse link function. The corresponding covariance matrix J_n is given by

$$J_n = \frac{1}{n} \sum_{i=1}^n h'(x_i^\top \beta)^2 w(\phi) V(\mu_i)^{-1} x_i x_i^\top. \quad (2.43)$$

In the following, we give explicit formulae for the empirical fluctuation processes in two important special cases of the GLM: the binomial (logistic) regression model and the log-linear poisson model.

Binomial model: Let y_i be the proportion of successes from m trials such that my_i is binomially distributed $\text{Bin}(\mu_i, m)$. Then the variance is determined by $w(\phi) = 1/m$ and $V(\mu) = \mu(1 - \mu)$ and if the canonical logit link is used then $h'(x) = \exp(x)/(1 + \exp(x))^2$. Given the ML estimates $\hat{\beta}$ and the corresponding fitted values $\hat{\mu}_i$, the covariance matrix for the M-fluctuation process can be estimated by

$$\hat{J} = \frac{1}{nm} \sum_{i=1}^n \frac{h'(x_i^\top \hat{\beta})^2}{\hat{\mu}_i(1 - \hat{\mu}_i)} x_i x_i^\top. \quad (2.44)$$

The empirical fluctuation process is then given by

$$efp(t) = \hat{J}^{-1/2} \frac{1}{\sqrt{n}} \sum_{i=1}^{\lfloor nt \rfloor} h'(x_i^\top \hat{\beta}) \frac{y_i - \hat{\mu}_i}{\hat{\mu}_i(1 - \hat{\mu}_i)} x_i. \quad (2.45)$$

Corresponding test statistics could be derived by, e.g., taking again the average Euclidian norm or the double maximum etc. Note that this methodology can also be applied if $m = 1$ where at each time i there is only one

observation of success ($y_i = 1$) or failure ($y_i = 0$). Applications of this process in a binomial model can be found in Section 2.6.2 and 2.6.3.

Poisson model: If the y_i are poisson distributed $\text{Poi}(\mu_i)$ then $V(\mu) = \mu$ and $w(\phi) = 1$. Using the canonical log link yields $h'(x) = \exp(x)$ so that the covariance can be estimated by

$$\hat{J} = \frac{1}{n} \sum_{i=1}^n \hat{\mu}_i x_i x_i^\top. \quad (2.46)$$

The empirical fluctuation process which is also given in Hjort and Koning (2002) simplifies to

$$efp(t) = \hat{J}^{-1/2} \frac{1}{\sqrt{n}} \sum_{i=1}^{\lfloor nt \rfloor} (y_i - \hat{\mu}_i) x_i \quad (2.47)$$

and test statistics can be computed like above. As pointed out for the OLS-based CUSUM test in Section 2.5.1 a test for changes in the mean the first component of the non-decorrelated process can also be used alone which has to be standardized by element (1,1) of the covariance matrix. For the poisson model this gives

$$efp(t) = \frac{1}{\sqrt{n}} \sum_{i=1}^n \frac{y_i - \hat{\mu}_i}{\sqrt{\bar{\mu}}}, \quad (2.48)$$

where $\bar{\mu}$ is the arithmetic mean of the fitted values $\hat{\mu}_i$. If the variances are constant, i.e., if there is only a constant regressor $x_i \equiv 1$ this is a CUSUM process based on the Pearson residuals. If not, it is almost a Pearson residuals CUSUM process except that the variance is estimated by $\bar{\mu}$ rather than $\hat{\mu}_i$. The latter is not possible if $\hat{\mu}_i$ is not consistent for the asymptotic variance, that is, the variance has to be estimated from a set of observations and the size of this set has to go to infinity with n . This Pearson residual-based CUSUM process is applied to a Poisson model in Section 2.6.4.

With a simple modification both processes can not only be used in poisson but also in quasi poisson models where overdispersion is allowed.

Then, the variance is not required to be equal to the mean but can be $\text{VAR}[y_i] = \phi\mu_i$. Note that in this case the density function is not given by (2.40) with $w(\phi) = \phi$. The dispersion parameter is a nuisance parameter and can be consistently estimated by $X^2/(n - k)$, where X^2 is the usual Pearson χ^2 statistic. To obtain properly standardized fluctuation processes, $efp(t)$ from (2.47) or (2.48) respectively has to be multiplied by $1/\sqrt{\hat{\phi}}$ which can then be used as usual for testing the constancy of the regression coefficients β .

2.5.3 Dependent Data

As stated above, the assumption of independent observations is often (but not necessarily) violated, in particular when dealing with time series data. Several approaches are conceivable when the methodology introduced above is to be applied to dependent data.

When using ML estimation techniques the parameters can be estimated from a fully specified likelihood or from a conditional likelihood and the fluctuation processes can be derived accordingly. But in many situations this is not necessary as consistent estimates $\hat{\theta}$ (or $\hat{\beta}$ in regression frameworks) can be obtained from the usual estimating equations (Godambe 1985; Liang and Zeger 1986). But as Lumley and Heagerty (1999) point out, it is crucial for inference in such models to compute consistent estimates for the covariance matrix \hat{B} (or \hat{J} respectively in regression models). Lumley and Heagerty (1999) suggest a class of weighted empirical adaptive variance estimators which are consistent in the presence of correlation in the data. These can be plugged into the fluctuation processes described above which renders the asymptotic theory valid again.

2.6 Applications

We illustrate a few of the tests discussed above by applying them to the following four models: an error correction model (ECM) for German M1 money demand, a binomial GLM for the fraction of illegitimate births in Großarl and for simulated binary data and a Poisson model for the number of youth homicides in Boston. The three “real world” data sets are included in the package `strucchange` implemented in the R system for statistical computing which are both presented in much more detail in the following chapter.

2.6.1 German M1 Money Demand

Lütkepohl, Teräsvirta, and Wolters (1999) investigate the stability and linearity of a German M1 money demand function and find a stable relationship for the time before the German monetary unification on 1990-06-01 but a clear structural change afterwards. They used seasonally unadjusted quarterly data from 1961(1) to 1995(4) for the logarithm of real M1 per capita m_t , the logarithm of a price index p_t , the logarithm of the real per capita gross national product y_t and the long-run interest rate R_t . The data were originally provided by the German central bank and are now available on the World Wide Web in the data archive of the *Journal of Applied Econometrics* (<http://qed.econ.queensu.ca/jae/1999-v14.5/lutkepohl-terasvirta-wolters/>).

Lütkepohl et al. (1999) used smooth transition regression to model the parameter instability; Zeileis et al. (2004) discuss this model in a structural change framework, but only based on OLS residuals and estimates not based on M-scores. We use the adapted model of Zeileis et al. (2004)

for the German M1 money demand to test the stability of the full sample estimates. Figure 2.1 shows the L_2 norm of the score-based fluctuation process defined by (2.37) and (2.38) as discussed in Section 2.5.1. The dashed horizontal line represents the mean L_2 norm $\|efp(t)\|_2^2$, i.e., the test statistic of the Nyblom-Hansen test, which exceeds its 5% critical value (solid line). Additionally to the information that the test finds evidence for structural change in the data, the clear peak in the fluctuation process conveys the information that the break seems to have occurred in about 1990, corresponding to the German monetary unification (highlighted by the dotted vertical line). The corresponding p value is 0.022.

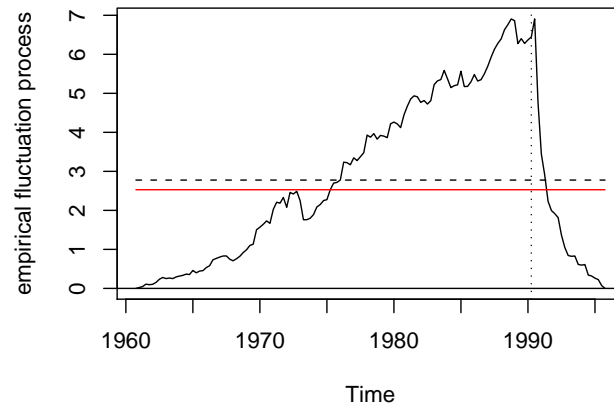


Figure 2.1: Score-based fluctuation process (mean L_2 norm) for German M1 data

2.6.2 Illegitimate Births in Großarl

In 18th century Salzburg, Austria, the reproductive behaviour was confined to marital unions due to sanctions by the catholic church and the legal system. Nevertheless, illegitimate births happened although the catholic church tried to prevent them by moral regulations of increasing severity. [Veichtlbauer, Hanser, Zeileis, and Leisch \(2002\)](#) discuss the impact of these and other policy interventions on the population system in Großarl, a small village in the Austrian Alps in the region of the archbishopric Salzburg. [Zeileis and Veichtlbauer \(2002\)](#) model the structural breaks in the annual fraction of illegitimate births (see [Figure 2.2](#)) by means of OLS.

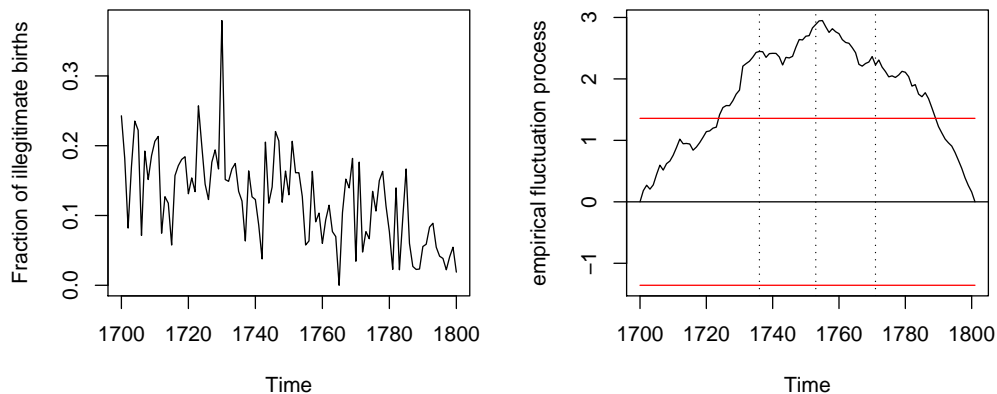


Figure 2.2: Illegitimate births in Großarl and the binomial CUSUM process

Here, we discuss the number of illegitimate and legitimate births between 1700 and 1800 in a binomial regression framework which is more appropriate for this kind of data (although the fitted values are equivalent for a regression on a constant). There were about 55 births per year in Großarl

during the 18th century—about seven of which were illegitimate—with an IQR of (48, 63). During this time the close linkage between religiosity and morality and between church and state led to a policy of moral suasion and social disciplining, especially concerning forms of sexuality that were not wanted by the catholic church. Moral regulations aimed explicitly at avoiding such unwanted forms of sexuality, e.g., by punishing fornication by stigmatising corrections, corporal punishment, compulsory labour or workhouse-prison. Women sometimes even had to leave the court district afterwards to avoid recidivism. After secularisation such regulations were abolished in the 19th century. To assess whether such interventions have any effect on the mean fraction of illegitimate births we employ the CUSUM test based on ML-scores from a binomial model as defined in Equation (2.45). The resulting empirical fluctuation process in Figure 2.2 clearly exceeds its boundary and therefore provides evidence for a decrease in the fraction of illegitimate births suggesting that the moral regulations have been efficient. The peak in the process conveys that there has been at least some structural break at about 1750, but two minor peaks on the left and the right can also be seen in the process. These match very well with the three major moral interventions in 1736, 1753 and 1771 respectively (indicated by dotted lines). The corresponding p value is < 0.0001 .

2.6.3 Artificial Binary Data

To show that the approach of M-fluctuation processes is not only applicable in situations like above where also OLS estimation techniques could be used despite a binomial GLM being more appropriate, we analyse an artificial data set of binary observations with covariates. We simulate $n = 200$ observations from a binomial GLM as described in Section 2.5.2 with the

canonical logit link. At each time i the response variable is only $m = 1$ observation of success ($y_i = 1$) or failure ($y_i = 0$) and the vector of covariates is $x_i = (1, (-1)^i)^\top$. A single shift model with changepoint $t = 0.5$ is used, i.e., 100 observations in each segment, and the vector of regression coefficients in segment 1 is $\beta_A = (1, 1)^\top$ which changes to $\beta_B = (0.2, 1)^\top$ in segment 2. Thus, the model corresponds to alternating success probability $\mu_i = 0.5$ and 0.881 in the first segment which drop to alternating success probabilities of 0.31 and 0.769 in the second segment. As only the first regression coefficient but not the remaining one changes this type of alternative is also called partial structural change. Unlike the previous example, the inspection of the raw time series data in Figure 2.3 does not shed much light on whether or not there has been a change in the parameters of the underlying model.

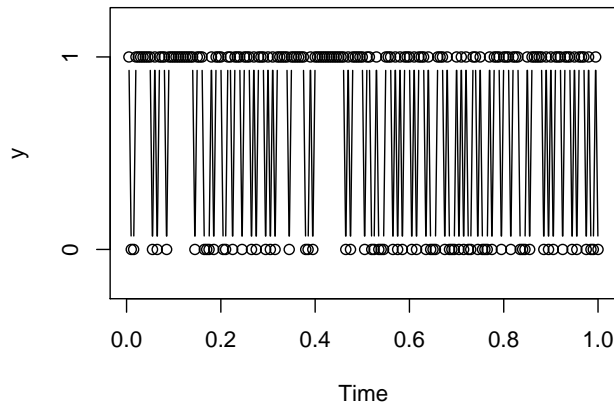


Figure 2.3: Artificial binary data

However, if the empirical fluctuation process from Equation (2.45) is derived the structural instability can clearly be seen as in Figure 2.4. It de-

picts the 2-dimensional fluctuation process with the boundary and process for the intercept in the upper panel and the process for the covariate $(-1)^i$ in the lower panel. This corresponds to using the double max statistic from (2.39) which allows for both identification of the instable parameter and the timing of the shift as discussed in Section 2.4.2. As only the first process crosses its boundary the test is able to pick up that the partial break is only associated with the intercept, while the moderate fluctuation of the second process reflects that the corresponding regression coefficient remains constant. The peak in the middle of sample period matches the true breakpoint of $t = 0.5$ (dotted line) very well. In addition, the fluctuation processes in Figure 2.4 illustrate that although the response variable is just binary the functional limit theorem works very well. The p value

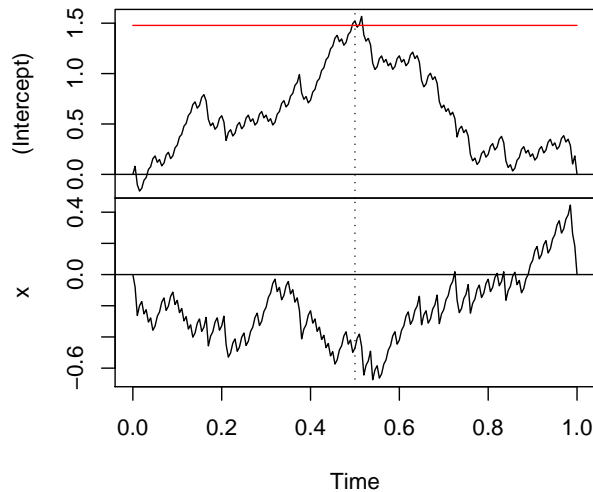


Figure 2.4: 2-dimensional binomial CUSUM process

corresponding to the double max test is 0.029.

Further discussion of structural changes in historic demographic time series of births and deaths can be found in [Veichtlbauer et al. \(2002\)](#) and [Zeileis \(2001\)](#).

2.6.4 Boston Homicide Data

To address the problem of continuing high homicide rates in Boston, in particular among young people, a policing initiative called the “Boston Gun Project” was launched in early 1995. This project implemented what became known as the “Operation Ceasefire” intervention in the late spring of 1996 which aimed at lowering homicide rates by a deterrence strategy. More information about youth homicide in Boston can be found in [Kennedy, Piehl, and Braga \(1996\)](#). As a single shift alternative seems reasonable but the precise start of the intervention cannot be determined, [Cooper, Piehl, Braga, and Kennedy \(2001\)](#) chose to model the number of youth homicides per month in Boston (see [Figure 2.5](#)) using modifications of the F tests for structural change of [Andrews \(1993\)](#) and [Andrews and Ploberger \(1994\)](#) assessing the significance via Monte Carlo results. In their regression model they include control variables like the population or a factor coding the month, but both have no significant influence at a 10% level. Hence, we use a much simpler model with a straightforward corresponding test: as is natural for count data we model the mean of the number of homicides by a Poisson model and assess the stability of the mean using the CUSUM process of the Pearson residuals as defined in [Equation \(2.48\)](#).

The corresponding empirical fluctuation process can be seen on the right

in Figure 2.5. As the process crosses its boundary (i.e., the L_∞ norm is used) there is evidence for a decrease of the number of homicides. Furthermore, the peak in the process indicates that the change seems to have occurred around early 1996 when the Operation Ceasefire was implemented (dotted line). The corresponding p value is < 0.0001 .

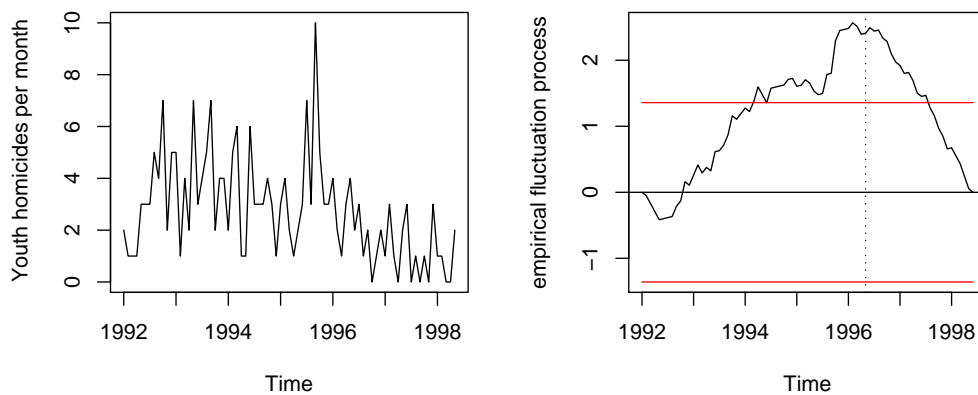


Figure 2.5: The Boston homicide data and the Poisson CUSUM process

2.7 Conclusions

This chapter proposes a general class of tests for parameter instability based on partial sum fluctuation processes of M-estimation scores. Based on functional central limit theorems for these fluctuation processes both under the hypothesis and local alternatives it is shown how structural changes in parametric models, with a special emphasis on regression models, can be discovered by test statistics that capture the fluctuation in the M-score processes.

The strategy derived can be summarized as follows: Given a data set, choose a model which should be tested for parameter stability or in which structural changes should be revealed, and choose an estimation technique suitable for the data under consideration. From the estimation technique the choice of the scores ψ follows naturally yielding an empirical fluctuation process. To check for excessive fluctuation in this process a functional has to be chosen which brings out either the timing of a structural change or the component incorporating the instability or both. For many such functionals the resulting significance test can be enhanced by a visualization method that not only displays the result of the test procedure but also conveys information about the type of structural instability and thus allows for better understanding the structure of the data.

Chapter 3

Implementation of Tests for Structural Change in the R Package `strucchange`

3.1 Introduction

This chapter concerns ideas and methods for implementing tests for structural change in linear regression relationships in a comprehensive and flexible way, that reflects the common features of the testing procedures. It offers facilities to display the results in various ways and unifies the testing approaches from the generalized M-fluctuation framework introduced in the previous chapter which extends the class of generalized fluctuation tests ([Kuan and Hornik 1995](#)) and from the F test framework ([Andrews 1993](#); [Andrews and Ploberger 1994](#)). The generalized fluctuation tests include tests based on residuals like the CUSUM ([Brown et al. 1975](#); [Ploberger and Krämer 1992](#)) and MOSUM ([Bauer and Hackl 1978](#); [Chu](#)

et al. 1995a) tests, based on parameter estimates (Ploberger et al. 1989; Chu et al. 1995b) or based on M-estimating scores as discussed in Chapter 2. The Chow (Chow 1960) and the $\text{sup}F$ test (Andrews 1993) belong to the F test framework. 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 (Chu, Stinchcombe, and White 1996; Leisch, Hornik, and Kuan 2000; Zeileis et al. 2004). All these methods have in common that either an empirical fluctuation process or a sequence of F statistics can be computed, plotted together with appropriate boundaries and tested to assess the significance of the corresponding test.

This chapter is organized as follows: In Section 3.2 the standard linear regression model from Section 2.5.1 in the previous chapter is reconsidered and some notation is introduced which will be used to formulate the testing problem and the tests. Section 3.4 introduces a data set which is also available in the package and which is used for the examples in this chapter. The following sections 3.5, 3.6 and 3.7 will then explain the tests, how they are implemented in *strucchange* and give examples for each. Section 3.5 is concerned with computing empirical fluctuation processes, with plotting them and the corresponding boundaries and finally with testing for structural change based on these processes. Analogously, Section 3.6 introduces the F statistics and their plotting and testing methods before Section 3.7 extends the tools from Section 3.5 to the monitoring situation.

3.2 The Model

Reconsider the standard linear regression model from Section 2.5.1

$$y_i = x_i^\top \beta_i + u_i \quad (i = 1, \dots, n), \quad (3.1)$$

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. In this setup, tests for structural change are concerned with testing the null hypothesis of “no structural change”

$$H_0 : \beta_i = \beta_0 \quad (i = 1, \dots, n) \quad (3.2)$$

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.

As in the previous chapter, the assumptions on regressors and disturbances have to be such that they imply functional central limit theorems for the following fluctuation processes. As this has been discussed in Section 2.5.1 and is not part of the focus of this chapter, explicit assumptions are omitted here.

In what follows, $\hat{\beta}^{(i,j)}$ 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 usual 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 by $\hat{u}_i = y_i - x_i^\top \hat{\beta}^{(n)} = y_i - \hat{y}_i$ 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 for structural change are the recursive residuals

$$\tilde{u}_i = \frac{y_i - x_i^\top \hat{\beta}^{(i-1)}}{\sqrt{1 + x_i^\top \left(X^{(i-1)\top} X^{(i-1)} \right)^{-1} x_i}} \quad (i = k + 1, \dots, n), \quad (3.3)$$

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

3.3 The R System

The R system (<http://www.R-project.org/>) is a language and environment for statistical computing and graphics which is ‘not unlike’ the language S which was developed at Bell Laboratories (formerly AT & T, now Lucent Technologies) by John Chambers and colleagues (Becker, Chambers, and Wilks 1988; Chambers and Hastie 1992; Chambers 1998). There are two implementations of the S language: the commercial system S-PLUS and the open source system R (Ihaka and Gentleman 1996) which is available under the terms of the GNU General Public License (GPL) from the Comprehensive R Archive Network (CRAN) at <http://CRAN.R-project.org/>. R runs on all major platforms, including various UNIX flavours (including FreeBSD and Linux), Windows (9x/NT/2000/XP) and MacOS.

One of the biggest strengths of the R system is that it is highly extensible enabling users to supply their own code in the standardized form of packages which typically contain not only R code but also manual pages with executable examples and possibly data sets, additional documentation and code in other languages like C, C++ or FORTRAN which can

easily be interfaced for efficiency. R comes with powerful tools for construction, quality testing, building and installation of such packages. At the moment, more than 230 such packages—including the `strucchange` package—are available from CRAN. These supplement the core functionality of the base system which already includes many standard statistical techniques and supply methodology from all areas of modern statistics. A discussion of the suitability of R for econometric purposes can be found in [Cribari-Neto and Zarkos \(1999\)](#), and [Racine and Hyndman \(2002\)](#) describe how R can be used to teach econometrics, illustrating the usage of various packages including `strucchange`.

The open source philosophy combined with the flexibility and extensibility of the R system has led to a high participation and involvement of many developers around the world sharing their code and knowledge with the R community. Some insights into the research activities on R and on modern statistical computing in general can be found in the R newsletter ([Hornik and Leisch 2002](#)) or the proceedings of the workshop on “Distributed Statistical Computing” ([Hornik, Leisch, and Zeileis 2003](#)).

3.4 The Data

The data used for examples throughout this chapter are macroeconomic time series from the US. The data set contains the aggregate monthly personal income and personal consumption expenditures (in billion US dollars) between January 1959 and February 2001, which are seasonally adjusted at annual rates. This is available in `strucchange` and was originally taken from <http://www.economagic.com/>, a web site for economic times series. Both time series are depicted in [Figure 3.1](#).

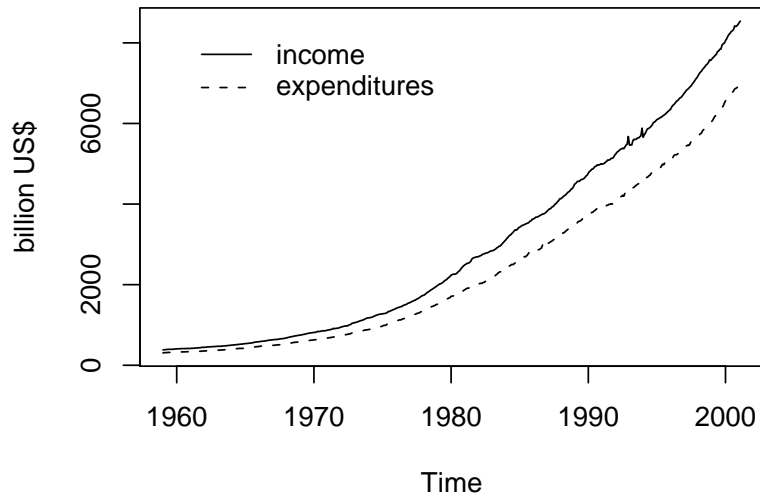


Figure 3.1: Personal income and personal consumption expenditures in the US

In R, the data can be loaded and a suitable subset chosen by

```
R> library(strucchange)
R> data(USIncExp)
R> library(ts)
R> USIncExp2 <- window(USIncExp, start = c(1985, 12))
```

We use a simple error correction model (ECM) for the consumption function similar to the model A of [Hansen \(1992\)](#):

$$\Delta ex_i = \beta_1 + \beta_2 r_{i-1} + \beta_3 \Delta in_i + u_i, \quad (3.4)$$

$$r_i = ex_i - \alpha_1 - \alpha_2 in_i, \quad (3.5)$$

where ex_i is the consumption expenditure and in_i the income. Alternatively to the model in levels like above, a model in logs could be used

which leads to similar results. We estimate the cointegration equation (3.5) by OLS and use the residuals \hat{r}_i as regressors in equation (3.4) in which we will test for structural change. Thus, the dependent variable is the increase in expenditure and the regressors are the cointegration residuals and the increments of income (and a constant). To compute the cointegration residuals and set up the model equation we need the following steps in R:

```
R> coint.res <- residuals(lm(expenditure ~ income,
  data = USIncExp2))
R> coint.res <- lag(ts(coint.res, start = c(1985, 12),
  freq = 12), k = -1)
R> USIncExp2 <- cbind(USIncExp2, diff(USIncExp2), coint.res)
R> USIncExp2 <- window(USIncExp2, start = c(1986, 1),
  end = c(2001, 2))
R> colnames(USIncExp2) <- c("income", "expenditure",
  "diff.income", "diff.expenditure", "coint.res")
R> ecm.model <- diff.expenditure ~ coint.res + diff.income
```

The objects `expenditure ~ income` and `ecm.model` are so-called formulas which define a regression model like (3.1). This is a version of the notation introduced by [Wilkinson and Rogers \(1973\)](#), as adapted for S ([Chambers and Hastie 1992](#)). The argument `data` specifies the data frame which actually contains the data for the variables.

Figure 3.2 shows the transformed time series necessary for estimation of equation (3.4).

In the following sections we will apply the methods introduced to test for structural change in this model.

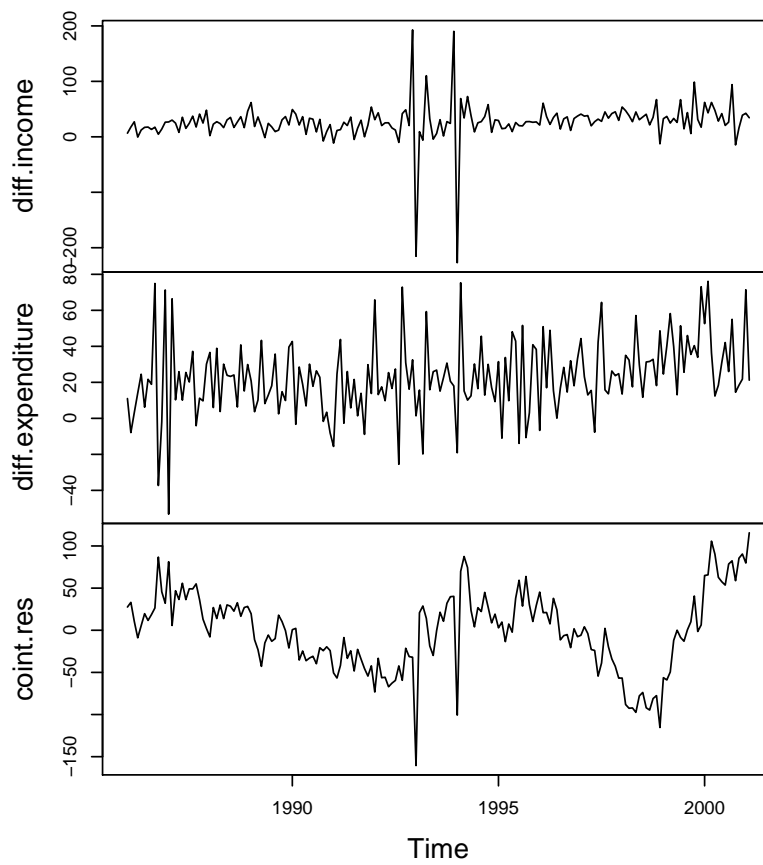


Figure 3.2: Time series used – first differences and cointegration residuals

3.5 Generalized Fluctuation Tests

The generalized fluctuation tests fit a model to the data and derive an empirical process that captures the fluctuation either in residuals, in estimates or in estimating scores. For these empirical processes (or possibly functionals of these) the limiting processes are known, so that boundaries can be computed, whose crossing probability under the null hypothesis is a prescribed significance level α . If the empirical process path crosses these boundaries, the fluctuation is improbably large and hence the null hypothesis should be rejected (at significance level α).

3.5.1 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 will be described in detail in this section.

Recursive residual-based processes: The first types of process that can be computed are CUSUM processes, which contain cumulative sums of standardized residuals. [Brown et al. \(1975\)](#) suggest cumulative sums of recursive residuals:

$$W_n(t) = \frac{1}{\tilde{\sigma}\sqrt{\eta}} \sum_{i=k+1}^{k+\lfloor t\eta \rfloor} \tilde{u}_i \quad (0 \leq t \leq 1), \quad (3.6)$$

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 as $n \rightarrow \infty$:

$$W_n(\cdot) \xrightarrow{d} W(\cdot). \quad (3.7)$$

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, Ploberger, and Alt \(1988\)](#) show that the main properties of the CUSUM quantity remain the same even under weaker assumptions, in particular in dynamic models. Therefore, `efp` has the logical argument `dynamic`; if set to `TRUE` the lagged observations y_{t-1} will be included as regressors.

Another possibility to detect a structural change is to analyze moving sums of residuals instead of cumulative sums. 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 along the whole sample period. Hence, the recursive MOSUM process ([Bauer and Hackl 1978](#)) is defined by

$$\begin{aligned} M_n(t|h) &= \frac{1}{\hat{\sigma}\sqrt{\eta}} \sum_{i=k+\lfloor N_\eta t \rfloor + 1}^{k+\lfloor N_\eta t \rfloor + \lfloor \eta h \rfloor} \tilde{u}_i \quad (0 \leq t \leq 1 - h) \quad (3.8) \\ &= W_n\left(\frac{\lfloor N_\eta t \rfloor + \lfloor \eta h \rfloor}{\eta}\right) - W_n\left(\frac{\lfloor N_\eta t \rfloor}{\eta}\right), \quad (3.9) \end{aligned}$$

where $N_\eta = (\eta - \lfloor \eta h \rfloor) / (1 - h)$. As the representation (3.9) suggests, the limiting process for the empirical recursive MOSUM process are the increments of a Brownian motion—this is shown in detail in [Chu et al. \(1995a\)](#).

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

Both processes are available in the function `efp` by specifying the argument type to be either "Rec-CUSUM" or "Rec-MOSUM", respectively.

OLS residual-based processes: [Ploberger and Krämer \(1992\)](#) suggested a structural change test based on cumulative sums of the standard OLS residuals instead of the recursive residuals. Thus, the OLS-CUSUM type empirical fluctuation process which was already discussed as a special case of the generalized M-fluctuation test in Section 2.5.1 is defined by:

$$W_n^0(t) = \frac{1}{\hat{\sigma}\sqrt{n}} \sum_{i=1}^{\lfloor nt \rfloor} \hat{u}_i \quad (0 \leq t \leq 1). \quad (3.10)$$

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 .

With the same ideas as for the recursive MOSUM test (3.8) the OLS-based MOSUM process is defined by

$$M_n^0(t|h) = \frac{1}{\hat{\sigma}\sqrt{n}} \sum_{i=\lfloor N_n t \rfloor + 1}^{\lfloor N_n t \rfloor + \lfloor nh \rfloor} \hat{u}_i \quad (0 \leq t \leq 1 - h) \quad (3.11)$$

$$= W_n^0\left(\frac{\lfloor N_n t \rfloor + \lfloor nh \rfloor}{n}\right) - W_n^0\left(\frac{\lfloor N_n t \rfloor}{n}\right), \quad (3.12)$$

where $N_n = (n - \lfloor nh \rfloor)/(1 - h)$. From (3.12) it can be seen that this process converges to the increments of a Brownian bridge which is also shown in [Chu et al. \(1995a\)](#).

Similar to the recursive MOSUM path this process should have a strong shift around t_0 under a single structural shift alternative.

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

Estimates-based processes: Instead of defining fluctuation processes on the basis of residuals they can be based equally well 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 former idea leads to the fluctuation process in the spirit of [Ploberger et al. \(1989\)](#) which is defined by

$$Y_n(t) = \frac{\sqrt{i}}{\hat{\sigma}\sqrt{n}} \left(X^{(i)\top} X^{(i)} \right)^{\frac{1}{2}} \left(\hat{\beta}^{(i)} - \hat{\beta}^{(n)} \right), \quad (3.13)$$

where $i = \lfloor k + t(n - k) \rfloor$ with $t \in [0, 1]$. And the latter gives the moving estimates (ME) process introduced by [Chu et al. \(1995b\)](#):

$$Z_n(t|h) = \frac{\sqrt{\lfloor nh \rfloor}}{\hat{\sigma}\sqrt{n}} \left(X^{\lfloor nt \rfloor, \lfloor nh \rfloor \top} X^{\lfloor nt \rfloor, \lfloor nh \rfloor} \right)^{\frac{1}{2}} \left(\hat{\beta}^{\lfloor nt \rfloor, \lfloor nh \rfloor} - \hat{\beta}^{(n)} \right), \quad (3.14)$$

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 \(1994\)](#) show 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="RE" the function `efp` returns the recursive estimates process, whereas for "ME" the moving estimates process is returned.

Score-based processes: As explained in detail in Chapter 2, a third possibility is to base fluctuation processes on sums of scores which are first order conditions (in an OLS framework) or ML scores (in a Gaussian model). The score at time i for component j is given by

$$\psi_{ij} = \begin{cases} (y_i - \hat{y}_i)x_{ij} & \text{for } j = 1, \dots, k, \\ (y_i - \hat{y}_i)^2 - \tilde{\sigma}^2 & \text{for } j = k + 1, \end{cases} \quad (3.15)$$

where $\tilde{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n \hat{u}_i^2$ is the ML estimate of the variance which is necessary for the empirical fluctuation process to return to 0 at $t = 1$. These scores are built for the extended model where also the error variance σ_i^2 may be time dependent and for the extended hypothesis that not only the regression coefficients are constant but also the variance. A test for the constancy of β alone could be based on the first k components of ψ_i as explained in Section 2.5.1. The covariance matrix estimate (see also (2.38)) is then given by

$$\hat{J} = \frac{1}{n} \sum_{i=1}^n \psi_i \psi_i^\top. \quad (3.16)$$

This leads to the following definition of the score-based CUSUM process

$$U_n(t) = \hat{J}^{-1/2} \frac{1}{\sqrt{n}} \sum_{i=1}^{\lfloor nt \rfloor} \psi_i. \quad (0 \leq t \leq 1), \quad (3.17)$$

which converges to a $k + 1$ -dimensional Brownian bridge as shown in Section 2.5.1. See also Hjort and Koning (2002) or Hansen (1992) for more

details.

Analogously to the ideas for the residual-based MOSUM tests, a score-based MOSUM process can be derived:

$$V_n(t|h) = \hat{f}^{-1/2} \frac{1}{\sqrt{n}} \sum_{i=\lfloor N_n t \rfloor + 1}^{\lfloor N_n t \rfloor + \lfloor nh \rfloor} \psi_i. \quad (0 \leq t \leq 1 - h) \quad (3.18)$$

$$= U_n \left(\frac{\lfloor N_n t \rfloor + \lfloor nh \rfloor}{n} \right) - U_n \left(\frac{\lfloor N_n t \rfloor}{n} \right), \quad (3.19)$$

which again converges to the increments of a Brownian bridge.

These processes can be computed by `efp` by setting `type` to "Score-CUSUM" or "Score-MOSUM", respectively. The processes should again have a peak or shift respectively under a single shift alternative.

To illustrate the use of `efp` we want to fit an OLS-based CUSUM process, and a moving estimates (ME) process with bandwidth $h = 0.2$ for our example data set. The commands are simply

```
R> ocus <- efp(ecm.model, type = "OLS-CUSUM", data = USIncExp2)
R> me <- efp(ecm.model, type = "ME", data = USIncExp2, h = 0.2)
```

These return objects of class "efp" which is a list containing mainly the empirical fluctuation processes and a few additional elements like the process type. The process itself is of class "ts" (the basic 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 $[0, 1 - h]$ as in (3.8) and (3.11).

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 for structural change. For the latter appropriate boundaries are needed; the concept of boundaries for fluctuation processes is explained in the next section.

3.5.2 Boundaries and Plotting

The idea 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 $efp(t)$ becomes improbably large compared to the fluctuation of the limiting process. The simplest way to check for excessive fluctuation is to reject the null hypothesis if at some time i some component $j = 1, \dots, \tilde{k}$ of the empirical fluctuation process ($efp_j(i/n)$) exceeds a boundary $b(t) = c \cdot d(t)$ that the limiting process just crosses with some probability α . In such a procedure, c determines the significance level α and $d(t)$ the shape of the boundary— \tilde{k} is 1 for the residual-based processes, k for the estimates-based processes and $k + 1$ for the score-based processes. This procedure corresponds to the test statistic (2.39) where the max functional is used for aggregating over both i and j and which allows for identification of both the timing of the shift and the component associated with the parameter instability.

The standard boundaries for most limiting processes have $d(t) = 1$, only for the Brownian motion it is $d(t) = 1 + 2t$ as this process has increasing variance. Whereas constant boundaries for the increments of a Brownian motion or bridge respectively are straightforward due to stationarity, both Brownian motions and Brownian bridges are neither variance nor covariance stationary and it seems plausible to remove at least the former non-standard feature by using alternative boundaries that are propor-

tional to the standard deviation: i.e., $d(t) = \sqrt{t}$ for the Brownian motion and $d(t) = \sqrt{t(1-t)}$ for the Brownian bridge. Zeileis (2004) discusses these boundaries in the context of residual-based CUSUM tests and shows that the alternative boundaries for Brownian bridges improve the power for structural changes early and late in the sample period—some further discussion of this topic can also be found in Zeileis et al. (2004).

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:

```
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. This is demonstrated in Figure 3.3.

It can be seen that the OLS-based CUSUM process exceeds its boundary; hence there is evidence for a structural change at the 5% level. Furthermore, the process seems to indicate two changes: one in the first half of the 1990s and another one at the end of 1998.

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

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

For multi-dimensional processes with $\tilde{k} > 1$ the `plot` method produces by default a one-dimensional process which has been computed by aggregating over j with the `max` functional. But as a crossing of the boundaries for

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

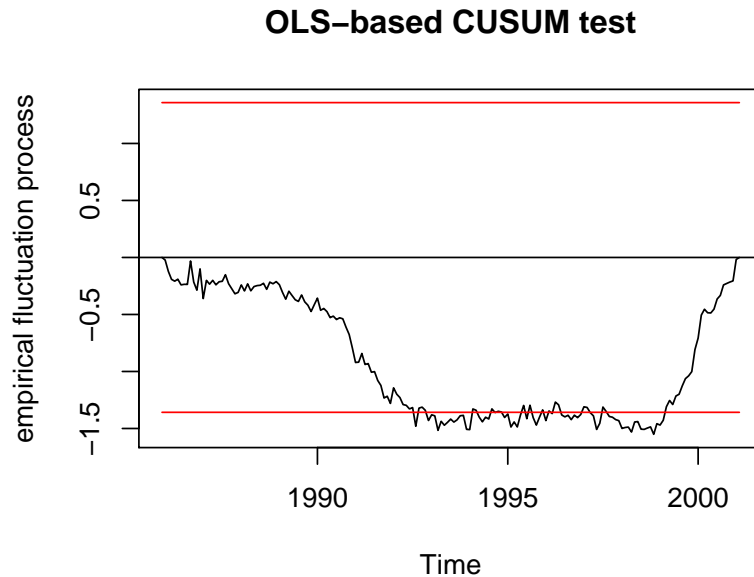


Figure 3.3: OLS-based CUSUM process

this process is equivalent to a crossing in some of the components of the unaggregated process, the \tilde{k} -dimensional process can also be plotted by setting the parameter `functional` (which defaults to "max") as `NULL`. The output from R can be seen in Figure 3.4, where the three parts of the plot show the processes that correspond to the estimate of the regression coefficients of the intercept, the cointegration residuals and the increments of income, respectively. All three paths show two shifts: the first shift starts at the beginning of the sample period and ends in about 1991 and the second shift occurs at the very end of the sample period. The shift that causes the significance seems to be the strong first shift in the process for the intercept and the cointegration residuals, because these cross their boundaries.

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

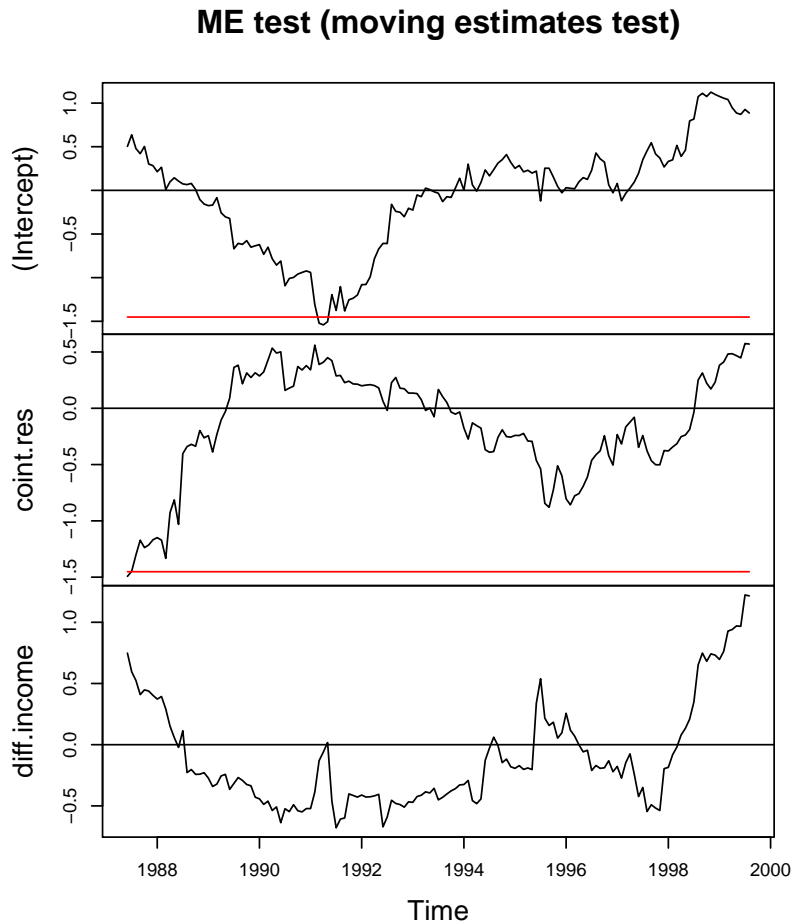


Figure 3.4: 3-dimensional ME process

Thus, the ME test leads to similar results as the OLS-based CUSUM test, but provides more information about the nature of the structural change. As pointed out in Section 2.4.2, other functionals for aggregating the empirical fluctuation process are conceivable and some of those allow for plotting in a very natural way. If in a first step a functional λ_{comp} is applied to $(efp_j(i/n))$ which aggregates over the components j the transformed

process $\lambda_{\text{comp}}(\text{efp}_j(i/n))$ could be plotted against time. Furthermore, if the functional λ_{time} which aggregates over time gives a scalar test statistic living on the same scale as the transformed fluctuation process it is sensible to add horizontal lines for the test statistic and the corresponding critical value as a boundary, indicating a significant result if the test statistic exceeds this boundary. If λ_{time} is the max functional this is again equivalent to the process itself crossing the boundary. In `strucchange` this can also be specified via the `functional` argument which, in addition to the default "max" functional, makes plots available for the "maxL2" and "meanL2" functional (for Brownian bridge type processes only), where in both cases λ_{comp} is the (squared) L_2 norm $\|\cdot\|_2^2$ and λ_{time} is the maximum

```
R> plot(ocus, functional = "meanL2")
```

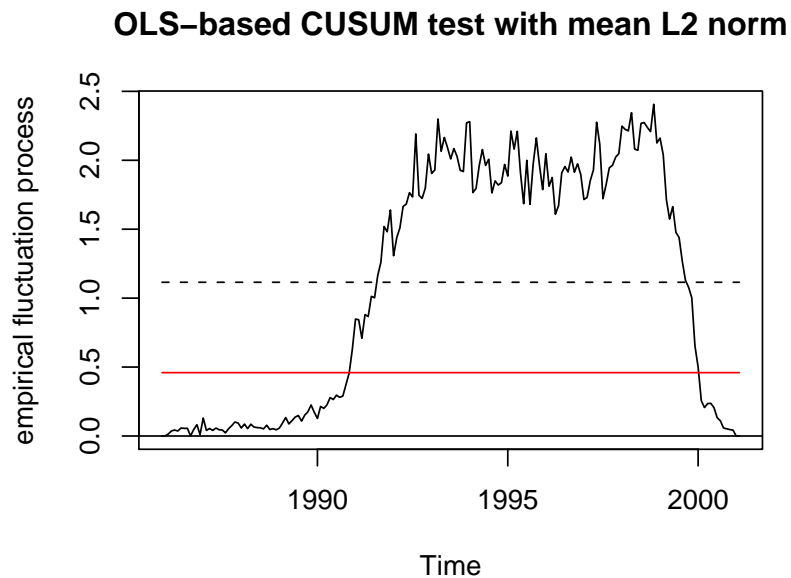


Figure 3.5: OLS-based CUSUM process with mean L_2 norm

or the arithmetic mean respectively. The latter is illustrated in Figure 3.5 which shows the transformed OLS-based CUSUM process together with its mean (dashed line) and the corresponding boundary. The interpretation is very much the same as for the process with the max functional: because the mean exceeds its boundary there is evidence for a structural change and the peaks in the process seem to indicate two shifts, one in about 1992 and one in about 1998.

Another functional λ_{time} which could be used for aggregating over time i is the range functional which is also available in *strucchange* but has no plotting method. It is available for significance testing, though, which is described in the following section.

3.5.3 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" (R's standard class for statistical test results) 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. In principle, many test statistics of type (2.28) are conceivable but in *strucchange* only the following ones are implemented:

$$\begin{aligned}
\text{functional} = \text{"max"} & \quad \max_{i=1, \dots, n} \max_{j=1, \dots, \tilde{k}} \left| \frac{efp_j(i/n)}{d(i/n)} \right| \\
\text{"maxL2"} & \quad \max_{i=1, \dots, n} \left\| efp_j(i/n) \right\|_2^2 \\
\text{"meanL2"} & \quad \frac{1}{n} \sum_{i=1}^n \left\| efp_j(i/n) \right\|_2^2 \\
\text{"range"} & \quad \max_{j=1, \dots, \tilde{k}} \text{range}_{i=1, \dots, n} \left| efp_j(i/n) \right|
\end{aligned} \tag{3.20}$$

The boundaries $d(t)$ are by default $d(t) = 1$ except for the Brownian motion, i.e., the recursive CUSUM test, $d(t) = 1 + 2t$. By setting `alt.boundary` to `TRUE` the alternative boundaries for the Brownian motion $d(t) = \sqrt{t}$ and for the Brownian bridge $d(t) = \sqrt{t(1-t)}$ can be used.

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)
```

and

```
R> sctest(ecm.model, type = "OLS-CUSUM", data = USIncExp2)
```

OLS-based CUSUM test

```
data: ecm.model
```

```
S0 = 1.5511, p-value = 0.01626
```

lead to equivalent results. `sctest` is a generic function which has methods not only for fluctuation tests, but all structural change tests (on historic data) introduced in this chapter including the F tests described in the next section.

3.6 F Tests

A rather 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 (3.1)

$$\beta_i = \begin{cases} \beta_A & (1 \leq i \leq i_0) \\ \beta_B & (i_0 < i \leq n) \end{cases}, \quad (3.21)$$

where i_0 is some change point in the interval $(k, n - k)$. [Chow \(1960\)](#) was the first to suggest such a test for structural change for the case where the (potential) change point i_0 is known. He proposed to fit two separate regressions for the two subsamples defined by i_0 and to reject whenever

$$F_{i_0} = \frac{\hat{u}^\top \hat{u} - \hat{e}^\top \hat{e}}{\hat{e}^\top \hat{e} / (n - 2k)}. \quad (3.22)$$

is too large, where $\hat{e} = (\hat{u}_A, \hat{u}_B)^\top$ are the residuals from the full model, where the coefficients in the subsamples are estimated separately, and \hat{u} are the residuals from the restricted model, where the parameters are fitted to all observations at once. The test statistic F_{i_0} has an asymptotic χ^2 distribution with k degrees of freedom and (under the assumption of normality) F_{i_0}/k has an exact 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 particular change point and which will be introduced in the following sections.

3.6.1 *F* Statistics: Function `Fstats`

A natural idea for extending 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 these statistics gets 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. These two parameters can not only be specified by the corresponding indices but alternatively also by fractions of the sample—the default is to take `from = 0.15` and implicitly `to = 0.85`—or by the corresponding dates as in the following example. To compute the *F* test statistics for all potential change points between January 1990 and June 1999 the appropriate command is:

```
R> fs <- Fstats(ecm.model, from = c(1990, 1),
               to = c(1999, 6), 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.

In addition to the arguments described above, `Fstats` takes an argument `cov.type` which allows for calculation of the *F* statistics based on heteroskedasticity robust covariance matrix estimates (the default is spherical errors).

3.6.2 Boundaries and Plotting

The computation of boundaries and plotting of F statistics is rather similar to that of empirical fluctuation processes introduced in the previous section. Under the null hypothesis of no structural change the sequence of F statistics converges to the square of a standardized tied-down Bessel process of order k (see [Andrews 1993](#), for details). Hence, as for the empirical fluctuation processes, boundaries can be computed such that the probability that the supremum (or the mean) of the limiting process exceeds this boundary is α . So the command

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

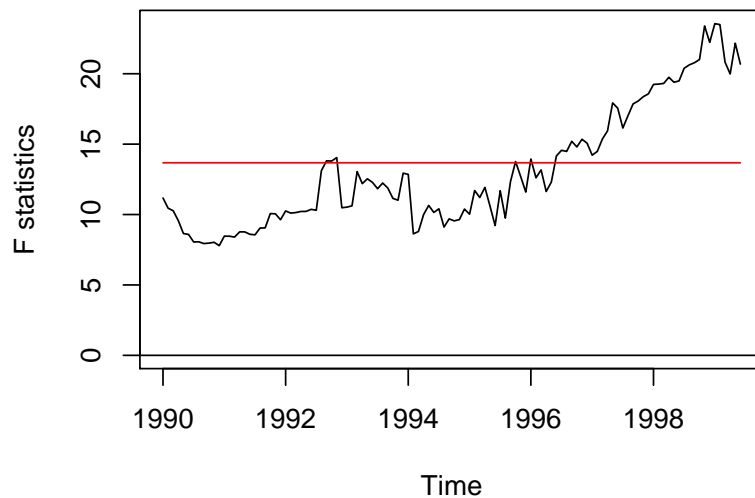


Figure 3.6: F statistics

plots the process of F statistics together with its boundary (for the supremum); the output can be seen in [Figure 3.6](#). As the F statistics cross their

boundary, there is evidence for a structural change (at the level $\alpha = 0.05$). The process has a clear peak in 1998, which mirrors the results from the empirical fluctuation processes and tests, respectively.

It is also possible to plot the corresponding p values by

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

which leads to equivalent results. In addition, 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.

3.6.3 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. [Andrews \(1993\)](#) and [Andrews and Ploberger \(1994\)](#) respectively suggest three different test statistics and examine their asymptotic distribution:

$$\text{sup}F = \sup_{\underline{i} \leq i \leq \bar{i}} F_i, \quad (3.23)$$

$$\text{ave}F = \frac{1}{\bar{i} - \underline{i} + 1} \sum_{i=\underline{i}}^{\bar{i}} F_i, \quad (3.24)$$

$$\text{exp}F = \log \left(\frac{1}{\bar{i} - \underline{i} + 1} \sum_{i=\underline{i}}^{\bar{i}} \exp(0.5 \cdot F_i) \right). \quad (3.25)$$

The $\text{sup}F$ statistic in (3.23) and the $\text{ave}F$ statistic from (3.24) respectively reflect the testing procedures that have been described above. The null

hypothesis is rejected when either the maximal or the mean F statistic becomes too large. A third possibility is to reject when the $\text{exp}F$ statistic from (3.25) becomes too large. The $\text{ave}F$ and $\text{exp}F$ test have certain optimality properties (Andrews and Ploberger 1994). The tests can be carried out in the same way as the fluctuation tests: either by supplying the fitted $F\text{stats}$ object or by a formula that describes the model to be tested. Hence, the commands

```
R> sctest(fs, type = "expF")
```

and

```
R> sctest(ecm.model, type = "expF", from = 49, to = 162,
         data = USIncExp2)
```

```
expF test
```

```
data: ecm.model
```

```
exp.F = 8.9955, p-value = 0.001311
```

produce equivalent output.

The p values are computed based on Hansen (1997).¹

3.7 Monitoring with the Generalized Fluctuation Test

The previous sections were concerned with the retrospective detection of structural changes in *given* data sets. Over the last years several fluctua-

¹We want to 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.

tion tests for structural change have been extended to monitoring of linear regression models where new data arrive over time (Chu et al. 1996; Leisch et al. 2000; Zeileis et al. 2004; Carsoule and Franses 2003). Such forward looking tests are closely related to sequential tests. When new observations arrive, the underlying fluctuation process is updated sequentially from all available data (historical plus newly arrived). As in the retrospective case, the hypothesis of no structural change is rejected if the fluctuation in the process becomes too large.

The standard linear regression model (3.1) is generalized to

$$y_i = x_i^\top \beta_i + u_i \quad (i = 1, \dots, n, n+1, \dots), \quad (3.26)$$

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 been developed for recursive (Chu et al. 1996) and moving (Leisch et al. 2000) estimates tests, for OLS-based CUSUM and MOSUM tests (Zeileis et al. 2004) and for ML scores in an AR(p) model (Carsoule and Franses 2003). 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 Brownian bridges there exists a closed form solution for boundary functions, such that the limiting 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 the increments of the Brownian bridge, only the growth rate of the boundaries can be derived analytically and critical values have to be simulated.

Consider that we want to monitor our ECM during the 1990s for structural change, using years 1986–1989 as the history period. First we cut the historic sample from the complete data set and create an object of class "mefp":

```
R> USIncExp3 <- window(USIncExp2, end = c(1989, 12))
R> me.mefp <- mefp(ecm.model, type = "ME", 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.

Assume new observations arrive for the year 1990. Calling function `monitor` on `me.mefp` automatically updates our monitoring object for the new observations stored in `USIncExp3` and runs a sequential test for structural change on each new observation (no structural break is detected in 1990):

```
R> USIncExp3 <- window(USIncExp2, end = c(1990, 12))
R> me.mefp <- monitor(me.mefp)
```

Then new data for the years 1991–2001 arrive and we repeat the monitoring:

```
R> USIncExp3 <- USIncExp2
R> me.mefp <- monitor(me.mefp)
```

Break detected at observation # 72

```
R> me.mefp
```

Monitoring with ME test (moving estimates test)

Initial call:

```
mefp.formula(formula = ecm.model, type = "ME",
             data = USIncExp3, alpha = 0.05)
```

Last call:

```
monitor(obj = me.mefp)
```

```
Significance level   : 0.05
Critical value       : 3.109524
History size         : 48
Last point evaluated : 182
Structural break at  : 72
```

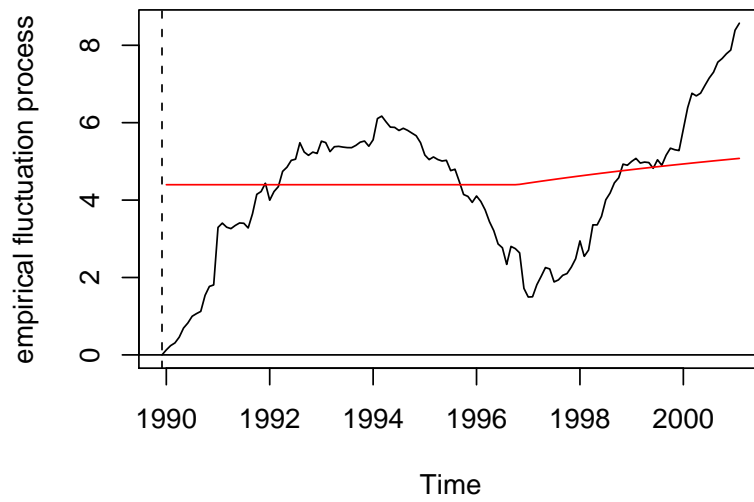
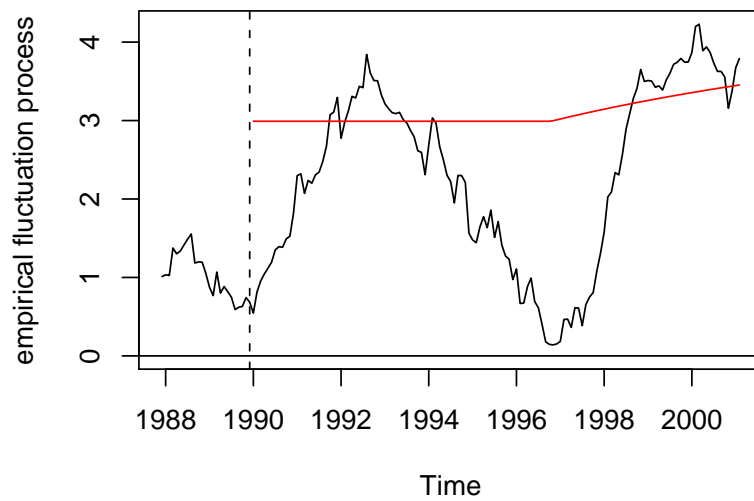
Parameter estimate on history :

```
(Intercept)  coint.res  diff.income
18.9299679  -0.3893141   0.3156597
```

Last parameter estimate :

```
(Intercept)  coint.res  diff.income
27.94869106  0.00983451   0.13314662
```

The software informs us that a structural break has been detected at observation #72, which corresponds to December 1991. 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 3.7 which shows the monitoring process with its

Monitoring with ME test (moving estimates test)Figure 3.7: Monitoring structural change with bandwidth $h = 1$ **Monitoring with ME test (moving estimates test)**Figure 3.8: Monitoring structural change with bandwidth $h = 0.5$

boundary and the start of the monitoring period as a vertical dashed line. 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(USIncExp2, end = c(1989, 12))
R> me.efp <- efp(ecm.model, type = "ME", data = USIncExp3,
  h = 0.5)
R> me.mefp <- mefp(me.efp, alpha = 0.05)
```

If now again the new observations up to February 2001 arrive, we can monitor the data

```
R> USIncExp3 <- USIncExp2
R> me.mefp <- monitor(me.mefp)
```

```
Break detected at observation # 70
```

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 do not have one history estimate that is being compared to the new moving estimates, but we have a history process, which can be seen on the left in Figure 3.8. This plot can simply be generated by `plot(me.mefp)`.

The results of the monitoring confirm the results of the historic tests: the moving estimates process has two strong shifts, the first around 1991 and the second around 1998.

3.8 Conclusions

This chapter has described the `strucchange` package that implements methods for testing for structural change in linear regression relationships. It

provides a unified framework for displaying information about structural changes flexibly and for assessing 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 for fitting empirical fluctuation processes (residual-, estimates- and score-based processes), computing an appropriate boundary, plotting these results and finally carrying 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- and OLS-based fluctuation processes have extensions to monitoring incoming data.

Given that a structural change (or possibly multiple structural changes) have been detected with the methods above another question arises: when did these breaks occur? The estimation of the corresponding breakpoints—also called dating of structural changes—and the determination of the number of breakpoints will be addressed in the next chapter.

Chapter 4

Testing and Dating of Structural Changes in Practice

4.1 Introduction

The previous chapters dealt with theory for tests for structural change and the implementation of classes of structural change tests in a software package. Both had some applications but these were chosen as illustrations for the theoretical properties and the implementation, respectively. In practice, the analysis of a data set does usually not stop after the detection of a structural change but continues with the modeling of this instability. Therefore, this chapter focuses on the application of structural change methods to regression models for several “real world” data sets including the following steps: testing for structural change, finding an appropriate segmentation for the data by estimating the number and timing of the breakpoints and fitting a segmented regression model.

Many classical methods for detecting and modeling structural changes in

linear regression models assume that there is just a single change of unknown timing under the alternative or that the timing and the type of change are known. More recently, there has been a surge of interest in methods which allow for multiple structural changes and in estimating the dates of shifts in such models, see [Bai \(1997b\)](#); [Hawkins \(2001\)](#); [Sullivan \(2002\)](#); [Bai and Perron \(2003\)](#), among many others. This chapter summarizes this more recent work and shows how to apply these methods and test for structural change in models with multiple breakpoints using the R package `strucchange`, thereby a bit rebalancing this literature, which is rather long on theoretical insights but often rather short on information on how these theories can be applied.

To illustrate this methodology, we use three data sets. The first one is the Nile data from [Cobb \(1978\)](#), later analyzed by [Dümbgen \(1991\)](#) and [Balke \(1993\)](#), which exhibits a level shift associated with the opening of the (first) Aswan dam at the end of the 19th century. The second data set is a time series of British road casualties analyzed by [Harvey and Durbin \(1986\)](#) which exhibits two breaks, one associated with increasing petrol prices in the wake of the first oil crisis, the second with the introduction of compulsory wearing of seatbelts in the early 1980s. Our third example—an index of the oil prices in Germany—was chosen in order to have a series with more than two breaks, in this case associated with events like the first oil crisis, the Iranian revolution, and the virtual breakup of OPEC in 1985. In Section [4.2](#) the notation for the linear regression model is extended to multiple change models and the methodology for dating these multiple changes is discussed. Section [4.3](#) describes how this methodology is implemented in the package `strucchange` which is then applied to the three data sets mentioned above in Section [4.4](#), [4.5](#) and [4.6](#) respectively.

4.2 Model and Methods

4.2.1 The Segmented Regression Model

The previous chapters discussed the standard linear regression model (3.1) in a structural change framework with time-dependent β_i . We introduced tests for the hypothesis of parameter stability $\beta_i \equiv \beta_0$ from (3.2) against the alternative that the coefficient β_i varies over time without further specifying the pattern of deviation from parameter stability under the alternative. Only for the F tests we argued that these are built for the single shift alternative (3.21) where the regression coefficients change only once at a (usually unknown) breakpoint. In many applications it is reasonable to assume that there is not only one breakpoint but that there are m breakpoints, where the coefficients shift from one stable regression relationship to a different one. Thus, there are $m + 1$ segments in which the regression coefficients are constant, and the model (3.1) can be rewritten as

$$y_i = x_i^\top \beta_j + u_i \quad (i = i_{j-1} + 1, \dots, i_j, \quad j = 1, \dots, m + 1), \quad (4.1)$$

where j is the segment index, $\mathcal{I}_{m,n} = \{i_1, \dots, i_m\}$ denotes the set of the breakpoints ($\mathcal{I}_{m,n}$ is also called m -partition), and by convention $i_0 = 0$ and $i_{m+1} = n$.

In practice, the breakpoints are rarely given exogenously but are unknown and have to be estimated from the data. This is what we do below. This methodology is valid under fairly general assumptions on regressors and disturbances, see e.g. Krämer et al. (1988) or Bai (1997b). Whereas the assumptions for structural change tests usually exclude trending regressors (except for some special cases such as the OLS-based CUSUM test with mean L_2 norm) the dating procedures discussed below are valid under

weaker assumptions. In particular trending regressors are permitted, see [Bai and Perron \(1998\)](#) for further discussion.

4.2.2 Testing Multiple Structural Changes

In principle, no particular tests are necessary for testing the null hypothesis of parameter constancy against the alternative of m breakpoints with $m > 1$. The generalized (M-)fluctuation tests from Chapter 2 and 3 do not assume a particular pattern of deviation from constancy anyway and the corresponding empirical fluctuation processes should exhibit increased fluctuation under the alternative in general. Also, the trajectories of the processes often shed light on the type of deviation from the null hypothesis such as the dating of the structural breaks. F statistics on the other hand are computed explicitly for a single shift alternative with $m = 1$. Under this alternative the process of F statistics should have a maximum close to the breakpoint i_1 . But as [Chong \(1995\)](#) showed, the process should have additional local maxima close to the other breakpoints if $m > 1$. This partial parameter consistency is illustrated in [Hansen \(2001\)](#). However, the power of the tests can be improved if similar F statistics are computed for multiple change alternatives. Therefore, [Bai and Perron \(1998, 2003\)](#) extend this approach to F tests for 0 versus ℓ breaks and ℓ versus $\ell + 1$ breaks respectively with arbitrary but fixed ℓ .

4.2.3 Dating Multiple Structural Changes

Given an m -partition i_1, \dots, i_m , the least squares estimates for the β_j can easily be obtained. The resulting minimal residual sum of squares is given

by

$$RSS(i_1, \dots, i_m) = \sum_{j=1}^{m+1} r_{ss}(i_{j-1} + 1, i_j), \quad (4.2)$$

where $r_{ss}(i_{j-1} + 1, i_j)$ is the usual minimal residual sum of squares in the j th segment. The problem of dating structural changes is to find the break-points $\hat{i}_1, \dots, \hat{i}_m$ which minimize the objective function

$$(\hat{i}_1, \dots, \hat{i}_m) = \operatorname{argmin}_{(i_1, \dots, i_m)} RSS(i_1, \dots, i_m) \quad (4.3)$$

over all partitions (i_1, \dots, i_m) with $i_j - i_{j-1} \geq n_h \geq k$. In applications, $n_h = \lfloor nh \rfloor$ will be a trimming parameter to be chosen by the practitioner. This is similar to the trimming of the F statistics as described in Section 3.6. Our examples below use $h = 0.1$ or $h = 0.15$.

Obtaining the global minimizers in (4.3) by an extensive grid search would be of order $O(n^m)$ and computationally burdensome for $m > 2$ (and any reasonable sample size n). Therefore, many hierarchical algorithms have been proposed that do recursive partitioning or joining of subsamples, see e.g. [Bai \(1997b\)](#) or [Sullivan \(2002\)](#), but these will not necessarily find the global minimizers. These can be found much easier by a dynamic programming approach that is of order $O(n^2)$ for any number of changes m . [Hawkins \(2001\)](#) discusses this for changepoint problems in a maximum likelihood framework which could also be extended to estimation of the parameters of a piecewise linear regression as suggested by [Hawkins \(1976\)](#). [Bai and Perron \(2003\)](#) present a version of that dynamic programming algorithm for pure and partial structural change models in an OLS regression context, which we adopt here. The basic idea is that of Bellman's principle: the optimal segmentation satisfies the recursion

$$RSS(\mathcal{I}_{m,n}) = \min_{mn_h \leq i \leq n - n_h} [RSS(\mathcal{I}_{m-1,i}) + r_{ss}(i + 1, n)]. \quad (4.4)$$

Therefore it suffices to know for each point i the “optimal previous partner” if i was the last breakpoint in an m -partition. This can be derived from a triangular matrix of $rss(i, j)$ with $j - i \geq n_h$, the computation of which is again made easier by the recursive relation $rss(i, j) = rss(i, j - 1) + \tilde{u}(i, j)^2$, where $\tilde{u}(i, j)$ is the recursive residual at time j of a sample starting at i (Brown et al. 1975). For more details on this dynamic programming algorithm see Bai and Perron (2003).

Bai and Perron (1998) also derive the rate of convergence which is n for the estimators of the break dates $\hat{\imath}_j/n$. To get an asymptotic distribution for the breakdates $\hat{\imath}_j$ themselves an asymptotic framework is adopted where the magnitudes of the structural shifts converge to zero as the sample size n increases. Based on this distribution, confidence intervals for the breakpoints can be computed; for details see Bai and Perron (1998, 2003).

4.3 Implementation in strucchange

Chapter 3 explained in detail how tests for structural change have been implemented in the R package `strucchange`. Here, we outline how we have incorporated methods in the package for dating multiple structural changes with unknown timing and multiplicity, fitting segmented regression models and flexibly displaying the results.

For the dating functions in `strucchange` an interface was chosen similar to that of the functions for fitting empirical fluctuation processes and sequences of F statistics (see Chapter 3):

```
efp(formula, data, type, ...)
Fstats(formula, data, cov.type, from = 0.15, ...)
```


where `formula` defines the regression model (3.1) and `data` is a data frame that might contain the corresponding variables and the argument. In `efp()` the argument `type` specifies the type of fluctuation process that should be fitted and in `Fstats()` the `cov.type` argument allows to calculate the F statistics based on heteroskedasticity robust covariance matrix estimates (the default is spherical errors) and `from` essentially specifies the trimming parameter h .

If there is evidence for structural changes in the regression relationship, these can be dated with the function

```
breakpoints(formula, data, breaks, h = 0.15, ...)
```

which implements the dynamic programming algorithm described above. In particular, it computes the triangular $rss(i, j)$ matrix. The parameter `breaks` is the number of breakpoints m , the default being the largest number m_h allowed by the trimming parameter h . From the object returned by this function any other number of breakpoints m can be extracted (as it contains the triangular $rss(i, j)$ matrix) by another application of the function `breakpoints` to this object. A summary of the fitted `breakpoints` object reports the breakpoints for $m + 1$ -segment models with $m = 0, \dots, m_h$ as well as the associated RSS and BIC . Such information criteria are often used for model selection, which in this case means selection of the number m of breakpoints. [Bai and Perron \(2003\)](#) argue that the AIC usually overestimates the number of breaks but that the BIC is a suitable selection procedure in many situations. The `plot` method for `breakpoint` objects plots both the RSS and the BIC for the various m -partitions and thus allows for visual determination of the model with the minimal BIC. A factor encoding the different regimes of the segmented regression can be extracted

using the function `breakfactor` which enables the user to easily fit the segmented regression model with optimally estimated breakpoints. Finally, the confidence intervals for some confidence level $1 - \alpha$ (which is 0.95 by default) can be extracted by the `confint` method. This will be illustrated in more detail in the applications in the following sections.

4.4 The Nile Data

First, we apply the above methods to the time series of the annual flow of the river Nile at Aswan from 1871 to 1970 (Cobb 1978; Dümbgen 1991; Balke 1993). It measures annual discharge at Aswan in 10^8 m^3 and is depicted in Figure 1.1 in Chapter 1.

We test whether the mean of the annual flow changes over time, i.e., we fit a constant to the data (inclusion of the lagged dependent variable does not lead to different results and was not found to have significant influence in the segmented model). In the S language this is written as `Nile ~ 1`, where `Nile` is a time series object containing the data. To check for structural changes in this model we first use the OLS-based CUSUM process from (3.10). The code below shows how to fit this fluctuation process and to produce the plot shown in Figure 4.1; it gives the process together with its boundaries at an (asymptotic) 5% significance level.

```
R> ocus.nile <- efp(Nile ~ 1, type = "OLS-CUSUM")
R> plot(ocus.nile)
```

The process has a peak around 1900 which exceeds the boundaries and hence indicates a clear structural shift at that time. The obvious reason is the Aswan dam that was built in 1898.

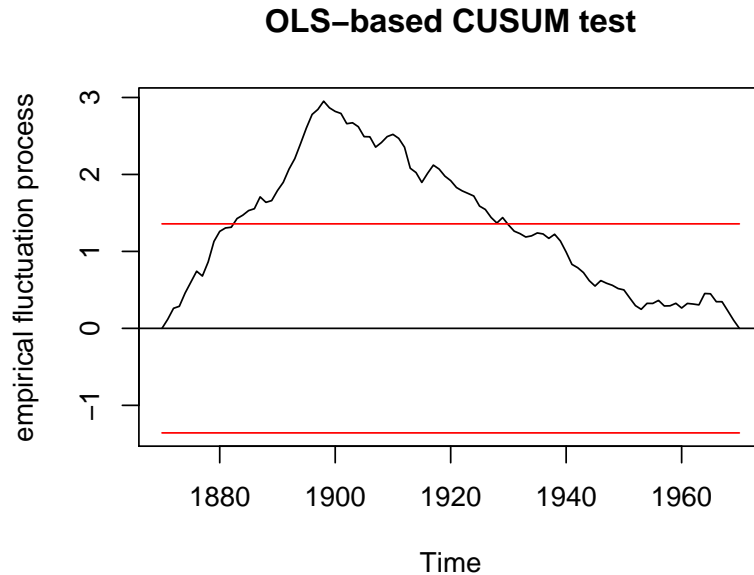


Figure 4.1: OLS-based CUSUM process for the Nile data

The same conclusion emerges from tests based on F statistics, as shown in Figure 4.2. The code below computes the F statistics and sets up a plot of the resulting process together with the boundaries corresponding to a $\text{sup}F$ test at the 5% significance level.

```
R> fs.nile <- Fstats(Nile ~ 1)
R> plot(fs.nile)
```

From this sequence of F statistics the optimal breakpoint for a 2-segment partition can be obtained as it is equivalent to maximizing the F statistic (3.22) or to minimizing the residual sum of squares (4.3). This breakpoint estimate can easily be obtained by `breakpoints(fs.nile)`. Although a 2-segment model seems quite intuitive for these data we also compare it to models with additional breakpoints. The following command computes

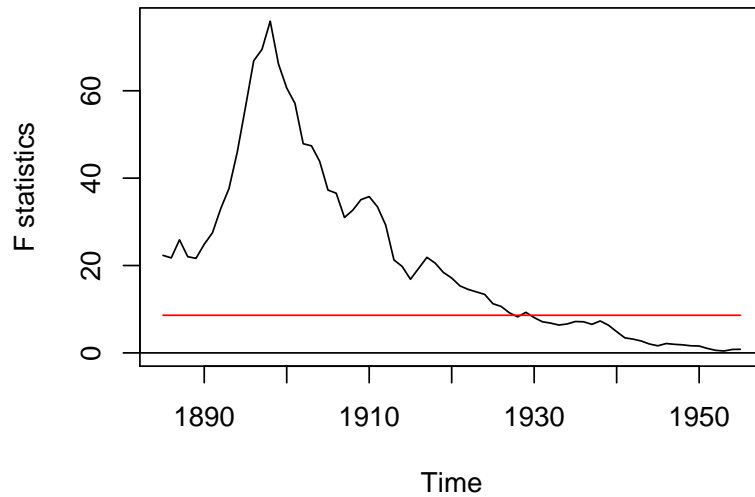


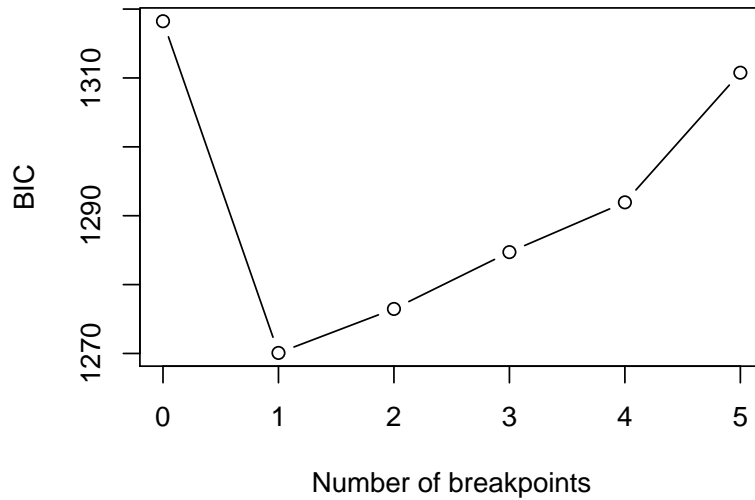
Figure 4.2: F statistics for the Nile data

arbitrary m -segment models based on the $rss(i, j)$ triangular matrix (with the default trimming of $h = 0.15$):

```
R> bp.nile <- breakpoints(Nile ~ 1)
```

A summary of this object reports the breakpoints for $m + 1$ -segment models with $m = 0, \dots, 5$ (the maximum possible with $h = 0.15$) as well as the associated RSS and BIC . Figure 4.3 shows that the BIC selects a model with $m = 1$ breakpoint, which confirms the results of the previous tests. The breakpoint for this model is observation 28—or equivalently the year 1898—and can be extracted by

```
R> bp1 <- breakpoints(bp.nile, breaks = 1)
```

Figure 4.3: BIC for models with m breakpoints

To summarize our results we fit two linear models to the data. The first model `fm0` is the model under the null hypothesis without any breaks and `fm1` is the estimated 2-segment model. The factor `nile.fac` is a suitable coding of the partition implied by the estimated breakpoint. It can be generated by the function `breakfactor` which draws on results of the `breakpoints` function. In the S language the function `lm` fits linear models whose coefficients can be extracted via `coef`.

```
R> fm0.nile <- lm(Nile ~ 1)
R> coef(fm0.nile)
```

```
(Intercept)
  919.35
```

```
R> nile.fac <- breakfactor(bp1)
R> fm1.nile <- lm(Nile ~ nile.fac - 1)
R> coef(fm1.nile)
```

```
nile.facsegment1 nile.facsegment2
      1097.7500      849.9722
```

The results can also be visualized as in Figure 4.4, which shows the fitted model for $m = 0, 1$ together with a vertical dashed line for the estimated breakpoint and its 90% confidence interval at the bottom. The steps for producing Figure 4.4 are outlined in the following lines of code, details like the legend, color and line type are omitted for simplicity.

```
R> plot(Nile)
R> lines(fitted(fm0.nile))
R> lines(fitted(fm1.nile))
R> lines(bp1)
R> lines(confint(bp.nile, breaks = 1, level = 0.9))
```

First, the time series itself is plotted, then the lines for the fitted model under the hypothesis `fm0.nile` and under the alternative `fm1.nile` are added and finally a vertical line for the estimated breakpoint and arrows for the confidence interval are drawn. The line `lines(bp1)` could also be omitted as it is included by default in the `lines` method for the `confint` object.

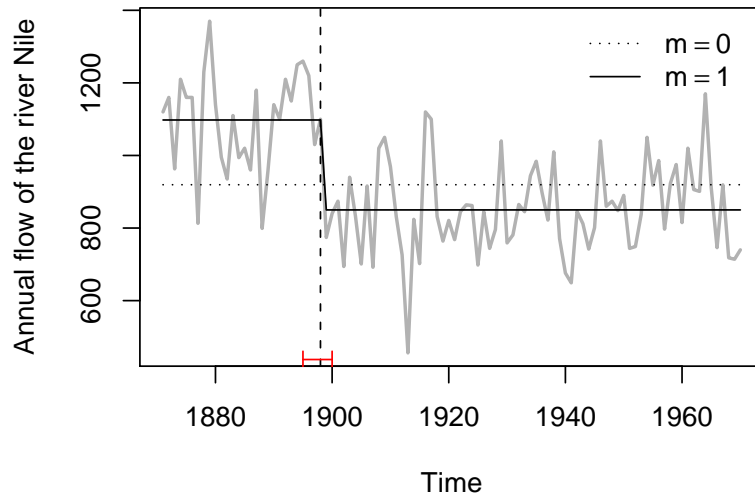


Figure 4.4: Fitted models for the Nile data

4.5 The Seatbelt Data

This section analyzes a monthly time series (from 1969(1) to 1984(12)) of the number of car drivers in Great Britain killed or seriously injured in traffic accidents ([Harvey and Durbin 1986](#)). An appropriate model for analyzing this time series in a least squares framework is to take the logarithm of the data and regress it on its lagged values at lag 1 and 12. This corresponds to a multiplicative SARIMA(1, 0, 0)(1, 0, 0)₁₂ model fitted by OLS. Another possibility would be to take first differences instead of logs which leads to very similar results, with slightly inferior fits.

The log-transformed series is depicted in [Figure 4.5](#).

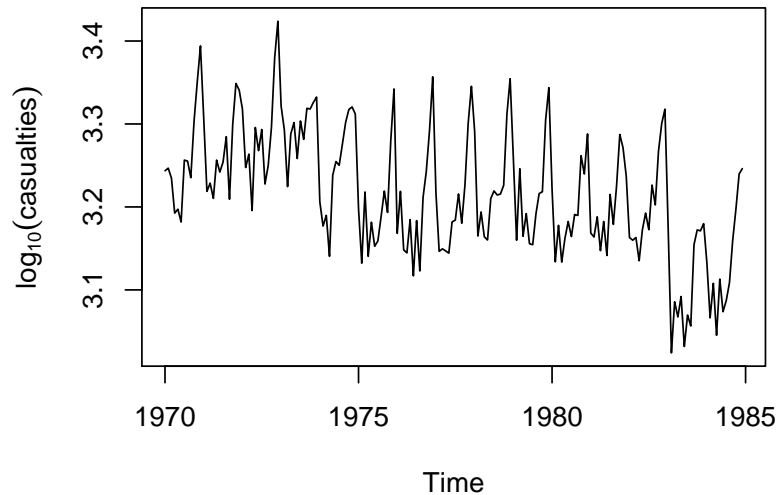


Figure 4.5: The seatbelt data

To test for structural changes we use the RE test from (3.13) and F statistics with a trimming parameter of $h = 0.1$. Both processes are fitted and plotted in the code given below and the results are shown in Figure 4.6.

```
R> re.seat <- efp(y ~ ylag1 + ylag12, data = seatbelt,
  type = "RE")
R> plot(re.seat, main = "RE test")
R> fs.seat <- Fstats(y ~ ylag1 + ylag12, data = seatbelt,
  from = 0.1)
R> plot(fs.seat, main = "supF test")
```

Both processes show significant departures from the null at an (asymptotic) 5% level, indicating at least one break in the data. But both processes have two clear peaks, the first in 1973(10) and the second in 1982(12) and

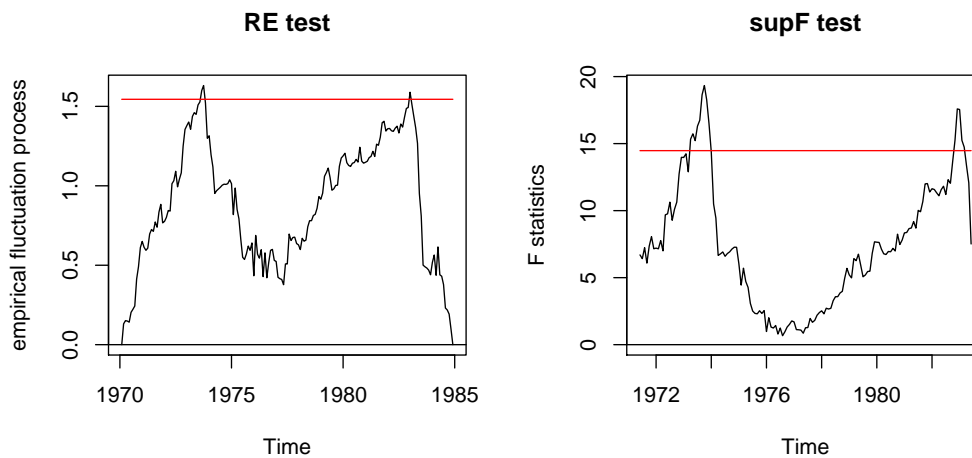


Figure 4.6: Structural change tests for the seatbelt data

1983(1) respectively, and favour a model with $m = 2$ breaks. To date the structural changes in this regression model we again estimate the break-points, with a trimming parameter of $h = 0.1$ and a maximum of $m = 5$ breaks.

```
R> bp.seat <- breakpoints(y ~ ylag1 + ylag12, data = seatbelt,
  h = 0.1, breaks = 5)
```

The BIC as shown in Figure 4.7 for the models with $m = 0, \dots, 5$ break-points would choose a model without any breaks, even though all the structural change tests above indicate at least one break. This might be caused by the fact that lagged regressors are included in which case the BIC might perform badly as pointed out by [Bai and Perron \(2003\)](#).

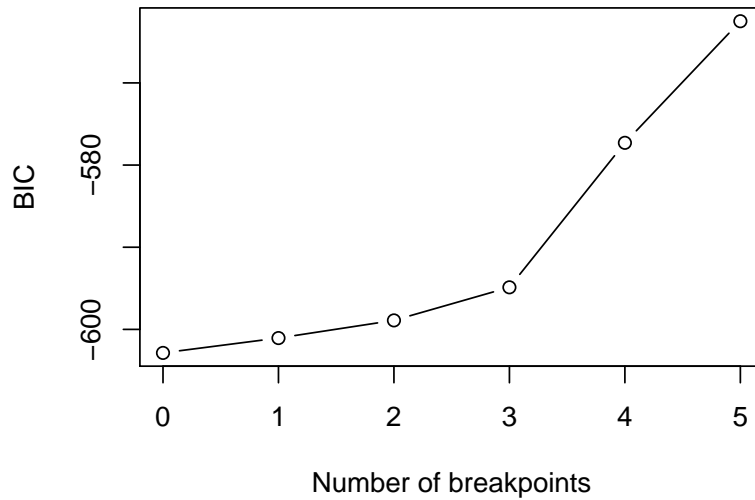


Figure 4.7: BIC for models with m breakpoints

Due to the two peaks in the recursive estimates process and the sequence of F statistics respectively we decide in favour of a model with two breakpoints:

```
R> bp2 <- breakpoints(bp.seat, breaks = 2)
```

The optimal breakpoints are then 1973(10)—associated with petrol rationing and the introduction of lower speed limits during the first oil crisis, see also the time series of oil prices in the following subsection—and 1983(1)—a level shift associated with the seat belt law introduced in the UK on 1983-01-31 (see [Harvey and Durbin 1986](#), for further details).

The fitted dependent variable with the two breaks can be seen in [Figure 4.8](#).

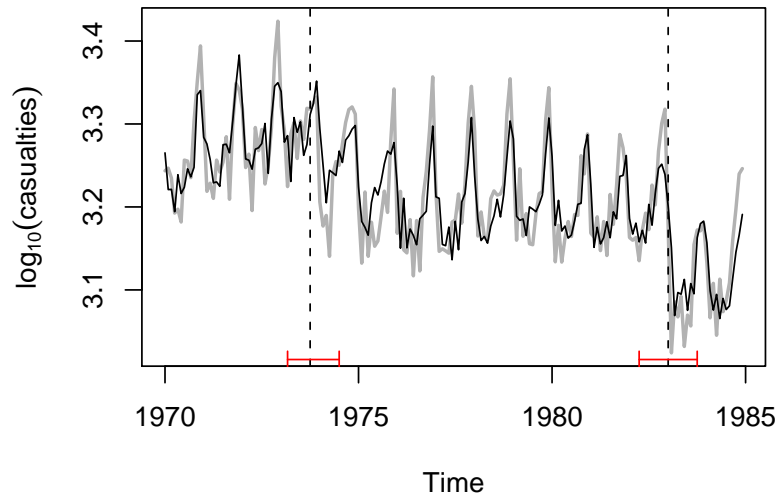


Figure 4.8: Fitted model for the seatbelt data

4.6 The Oil Price Data

This section considers a quarterly index of import prices of petroleum products—hereafter referred to as the oil price data—from 1960(1) to 1994(4) (base year: 1991). The data was obtained from the Statistisches Bundesamt Deutschland (Federal Statistical Office, Germany) and is given in Figure 4.9 (in logs).

Since there are obvious shifts in the mean, all structural change tests are highly significant. We therefore omit all details and directly proceed to estimating the breakpoints (with the default trimming of $h = 0.15$).

```
R> bp.oil <- breakpoints(log(OilPrice) ~ 1)
```

Again, a summary of this object would give information about the esti-

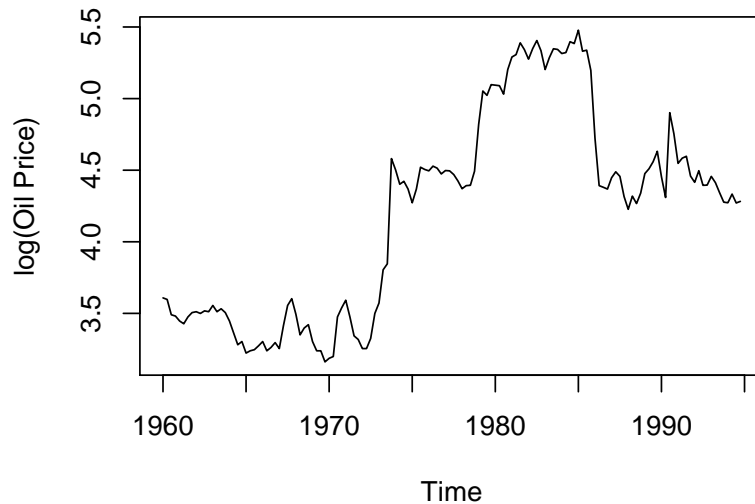


Figure 4.9: The oil price data

mated breakpoints and the associated RSS and BIC of partitions with $m = 0, \dots, 5$ breakpoints. For illustration, Figure 4.10 depicts the BIC , which is almost identical for 3 and 4 breaks. Hence, we first extract the segmentation with 3 breaks

```
R> bp3 <- breakpoints(bp.oil, breaks = 3)
```

and then we use the OLS-based CUSUM for checking for additional breaks:

```
R> ocus.oil <- efp(log(OilPrice) ~ breakfactor(bp3),
  type = "OLS-CUSUM")
R> plot(ocus.oil)
```

The CUSUM process in Figure 4.11 exhibits various peaks, the highest of which is in 1964, but none of these seems to be too extreme. So we stick to

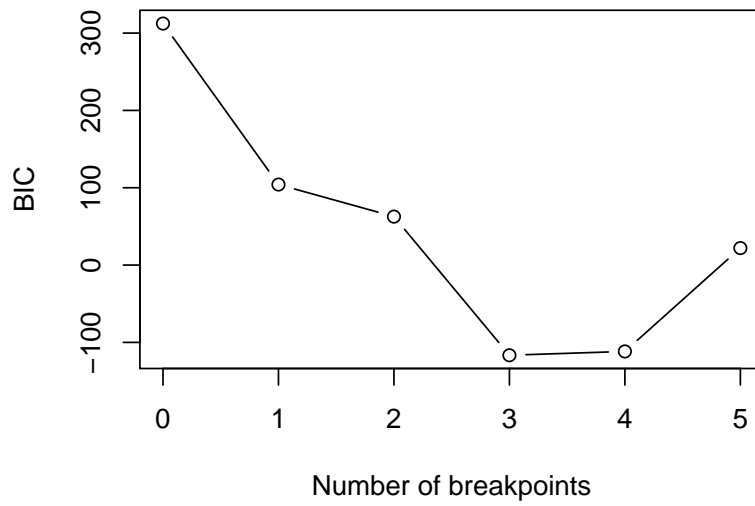
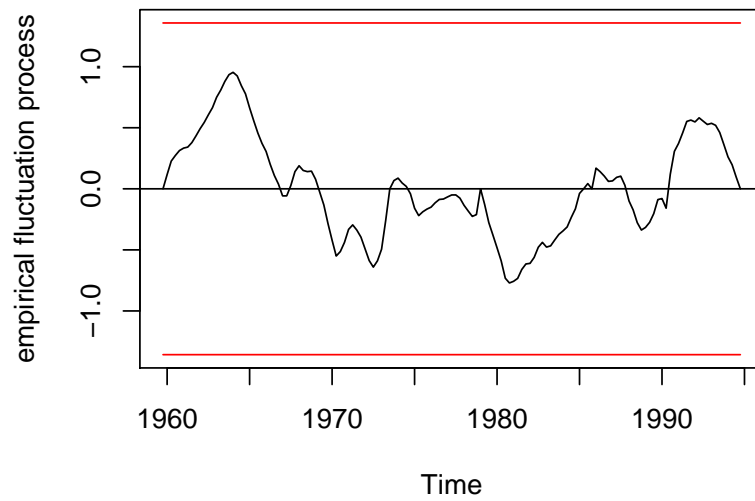
Figure 4.10: BIC for models with m breakpoints**OLS-based CUSUM test**

Figure 4.11: OLS-based CUSUM process for the oil price data

the model with 3 breakpoints which are 1973(3), 1979(1) and 1985(4). The fitted model is shown in Figure 4.12.

The period 1973(3), for quarterly data, is essentially the same breakpoint as 1973(10) for the monthly seatbelt data, which corresponds to the Arab oil embargo after the Yom Kippur war. 1979(1) marks the start of the Iranian revolution followed by the war between Iran and Iraq. The break in 1985(1) is, in hindsight, a joint product of various minor events such as a worldwide slowdown of demand, the entering of Great Britain, Norway and Mexico as major suppliers in international oil markets, and internal quarrels in the OPEC cartel, which led Saudi Arabia to increase its production and to abandon its role as the “residual supplier” which had until then softened all shocks in demand.

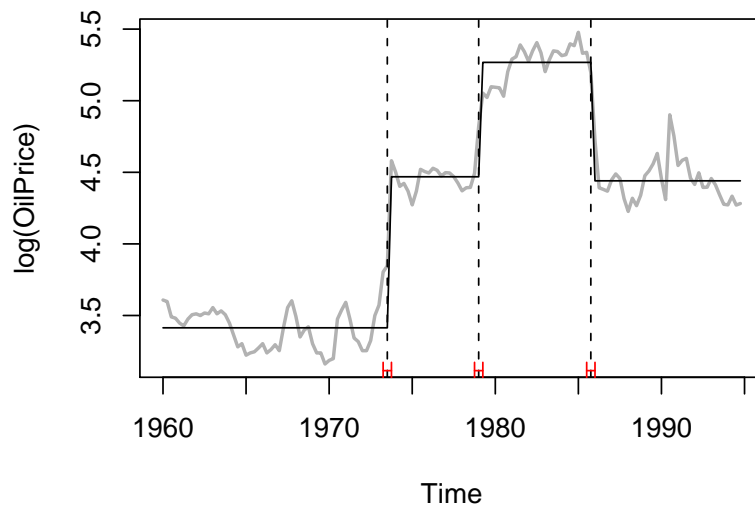


Figure 4.12: Fitted model for the oil price data

Chapter 5

Summary

Structural change is an important issue in many areas of research and data analysis. Tests for structural change are valuable tools both in diagnostic checking and in graphical exploration of the structure of the data. This monograph presents a theoretically and computationally sound and flexible framework for testing for structural change or parameter instability with a special emphasis on parameter instability in (generalized) linear regression models.

A generalized framework for testing for parameter instability based on partial sums of M-scores is introduced which uses and unifies the ideas from many tests for structural change known from the literature and generalizes them to various important estimation techniques for parametric models including ordinary least squares, maximum likelihood and M-estimation. These methods together with virtually all common tests for structural change in linear regression relationships are implemented in the R package `strucchange` which allows for computation, visualization and graphical analysis of empirical fluctuation processes and sequences of F statistics which often convey information about the presence and location

of breakpoints in the data as well as traditional significance testing. Furthermore, we have shown how recent methodological advances in testing against and dating multiple structural changes can be applied to “real” data. In summary, this gives an approach for testing for parameter instability from theory over implementation to application which allows for a lot of flexibility in each step of the structural analysis such that the procedure can be adjusted to the particular data set, but which always reflects the common features and properties so that a thread connects the steps of the analysis from model selection, estimation technique over capturing and visualizing information about potential structural changes to assessing their significance.

Although our results are very encouraging they should not conceal that there are many open research tasks which deserve further investigation. On the theoretical side, a few of the most interesting questions are:

- Given such a multiplicity of reasonable testing procedures, which performs best under a specific alternative? Although there are a few theoretical results such as those of [Nyblom \(1989\)](#) for random walk alternatives and of [Andrews and Ploberger \(1994\)](#) for single shift alternatives further insights for other types of departures from constancy are needed.
- When estimating the breakpoints, how can the number of breaks be determined by a significance procedure controlling the type I error? Currently, stepwise or recursive procedures and model selection based on information criteria can give rather unsatisfactory results as seen in [Chapter 4](#).
- How can boundaries for Brownian motions and bridges be chosen

more flexibly, e.g., such that the crossing probability is distributed according to a prior distribution? It is outlined that the weighting of fluctuation processes or equivalently using alternative boundaries can considerably affect the performance of the corresponding tests. If there is knowledge about the type or timing of a structural change this could be incorporated in suitable boundaries.

On the computational side, a more general implementation of the generalized M-fluctuation test framework is needed which allows for more flexible specification of ψ , λ and \hat{f} by the user. In particular, more support is needed for heteroskedasticity and autocorrelation consistent (HAC) covariance matrix estimation. These issues deserve further study and are currently under investigation.

Appendix 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 crossing probability of a Brownian motion for linear boundaries there is a rather good approximation to the asymptotic p value function given in [Zeileis \(2000\)](#). For the Brownian bridge with linear boundaries there is a series expansion given in [Ploberger and Krämer \(1992\)](#) which is evaluated for the first hundred terms. Similarly, for the range functional of a Brownian bridge there is a series expansion. If the max functional is used as λ_{comp} to aggregate over the components of a multi-dimensional Brownian bridge a Bonferroni approximation is used. For all other tests from the generalized fluctuation test framework the p values are computed by linear interpolation from simulated critical values. For the Brownian motion with alternative boundaries p values from the interval $[0.001, 1]$ and

[0.001, 0.999] for the Brownian motion respectively are approximated from tables given in [Zeileis \(2000\)](#). The critical values for the increments of a Brownian motion 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 increments of a Brownian bridge are given in [Chu et al. \(1995b\)](#); the parameter h is in both cases interpolated for values in [0.05, 0.5]. The critical values for the max and mean functional of the L_2 norm of Brownian bridge have been simulated in R for up to 25-dimensional Brownian bridges. If tabulated critical values are available for only a certain range of significance levels the approximation rule is the following: interpolate linearly between 1 and the largest available p value (e.g., 0.1) and for small p values give the smallest p value available (e.g., 0.01).

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.

Appendix B

strucchange Reference Manual

BostonHomicide *Youth Homicides in Boston*

Description

Data about the number of youth homicides in Boston during the ‘Boston Gun Project’—a policing initiative aiming at lowering homicide victimization among young people in Boston.

Usage

```
data(BostonHomicide)
```

Format

A data frame containing 2 monthly time series and a factor coding seasonality.

homicides time series. Number of youth homicides,

population time series. Boston population (aged 25-44), linearly interpolated from annual data.

season factor coding the month.

Details

The 'Boston Gun Project' is a policing initiative aiming at lowering youth homicides in Boston. The project began in early 1995 and implemented the so-called 'Operation Ceasefire' intervention which began in the late spring of 1996.

More information is available at:

<http://www.ksg.harvard.edu/criminaljustice/research/bgp.htm>

Source

Cooper et al. (2001), Figure 1 and Table I.

References

Cooper S.J., Piehl A.M., Braga A.A., Kennedy D.M. (2001), Testing for Structural Breaks in the Evaluation of Programs, Unpublished Paper, John F. Kennedy School of Government.

Kennedy D.M., Piehl A.M., Braga A.A. (1996), Youth Violence in Boston: Gun Markets, Serious Youth Offenders, and a Use-Reduction Strategy, *Law and Contemporary Problems*, 59, 147-183.

Examples

```
data(BostonHomicide)

fm <- glm(homicides ~ population + season, data = BostonHomicide,
          family = poisson)
anova(fm, test = "F")
```

Fstats

F Statistics

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 $(n*from) : (n*to)$, 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> . 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 <code>ts</code> , 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 "const" and White's heteroskedasticity consistent estimator is used if set to "HC". And "HC1" stands for a standardized estimator of "HC", 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.

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

```
require(ts)

## Nile data with one breakpoint: the annual flows drop in 1898
## because the first Ashwan dam was built
data(Nile)
plot(Nile)

## test the null hypothesis that the annual flow remains constant
## over the years
fs.nile <- Fstats(Nile ~ 1)
plot(fs.nile)
sctest(fs.nile)
## visualize the breakpoint implied by the argmax of the F statistics
plot(Nile)
lines(breakpoints(fs.nile))
```



```

## UK Seatbelt data: a SARIMA(1,0,0)(1,0,0)_12 model
## (fitted by OLS) is used and reveals (at least) two
## breakpoints - one in 1973 associated with the oil crisis and
## one in 1983 due to the introduction of compulsory
## wearing of seatbelts in the UK.
data(UKDriverDeaths)
seatbelt <- log10(UKDriverDeaths)
seatbelt <- cbind(seatbelt, lag(seatbelt, k = -1), lag(seatbelt, k = -12))
colnames(seatbelt) <- c("y", "ylag1", "ylag12")
seatbelt <- window(seatbelt, start = c(1970, 1), end = c(1984,12))
plot(seatbelt[, "y"], ylab = expression(log[10](casualties)))

## compute F statistics for potential breakpoints between
## 1971(6) (corresponds to from = 0.1) and 1983(6) (corresponds to
## to = 0.9 = 1 - from, the default)
## compute F statistics
fs <- Fstats(y ~ ylag1 + ylag12, data = seatbelt, from = 0.1)
## this gives the same result
fs <- Fstats(y ~ ylag1 + ylag12, data = seatbelt, from = c(1971, 6),
            to = c(1983, 6))
## plot the F statistics
plot(fs, alpha = 0.01)
## plot F statistics with aveF boundary
plot(fs, aveF = TRUE)
## perform the expF test
sctest(fs, type = "expF")

```

GermanM1

German M1 Money Demand

Description

German M1 money demand.

Usage

```
data(GermanM1)
```

Format

GermanM1 is a data frame containing 12 quarterly time series from 1961(1) to 1995(4) and two further variables. `historyM1` is the subset of GermanM1

up to 1990(2), i.e., the data before the German monetary unification on 1990-06-01. `monitorM1` is the complement of `historyM1`, i.e., the data after the unification. All three data frames contain the variables

- m** time series. Logarithm of real M1 per capita,
- p** time series. Logarithm of a price index,
- y** time series. Logarithm of real per capita gross national product,
- R** time series. Long-run interest rate,
- dm** time series. First differences of `m`,
- dy2** time series. First differences of lag 2 of `y`,
- dR** time series. First differences of `R`,
- dR1** time series. First differences of lag 1 of `R`,
- dp** time series. First differences of `p`,
- m1** time series. Lag 1 of `m`,
- y1** time series. Lag 1 of `y`,
- R1** time series. Lag 1 of `R`,
- season** factor coding the seasonality,
- ecm.res** vector containing the OLS residuals of the Lütkepohl et al. (1999) model fitted in the history period.

Details

Lütkepohl et al. (1999) investigate the linearity and stability of German M1 money demand: they find a stable regression relation for the time before the monetary union on 1990-06-01 but a clear structural instability afterwards.

Zeileis et al. (2002) use a model with `ecm.res` instead of `m1`, `y1` and `R1`, which leads to equivalent results in the history period but slightly different results in the monitoring period. The reason for the replacement is that stationary regressors are needed for the structural change tests. See references and the examples below for more details.

Source

The data is provided by the German central bank and is available online in the data archive of the Journal of Applied Econometrics <http://qed.econ.queensu.ca/jae/1999-v14.5/lutkepohl-terasvirta-wolters/>.

References

Lütkepohl H., Teräsvirta T., Wolters J. (1999), Investigating Stability and Linearity of a German M1 Money Demand Function, *Journal of Applied Econometrics*, **14**, 511-525.

Zeileis A., Leisch F., Kleiber C., Hornik K. (2002), Monitoring Structural Change in Dynamic Econometric Models, Report 64, SFB "Adaptive Information Systems and Modelling in Economics and Management Science", Vienna University of Economics, <http://www.wu-wien.ac.at/am/reports.htm#78>.

Examples

```
data(GermanM1)
## Lütkepohl et al. (1999) use the following model
LTW.model <- dm ~ dy2 + dR + dR1 + dp + m1 + y1 + R1 + season
## Zeileis et al. (2002) use
M1.model <- dm ~ dy2 + dR + dR1 + dp + ecm.res + season

## historical tests
ols <- efp(LTW.model, data = GermanM1, type = "OLS-CUSUM")
plot(ols)
re <- efp(LTW.model, data = GermanM1, type = "fluctuation")
plot(re)
fs <- Fstats(LTW.model, data = GermanM1, from = 0.1)
plot(fs)

## monitoring
M1 <- historyM1
ols.efp <- efp(M1.model, type = "OLS-CUSUM", data = M1)
newborder <- function(k) 1.5778*k/118
ols.mefp <- mefp(ols.efp, period = 2)
ols.mefp2 <- mefp(ols.efp, border = newborder)
M1 <- GermanM1
ols.mon <- monitor(ols.mefp)
ols.mon2 <- monitor(ols.mefp2)
plot(ols.mon)
lines(boundary(ols.mon2), col = 2)

## dating
bp <- breakpoints(LTW.model, data = GermanM1)
summary(bp)
plot(bp)

plot(fs)
lines(confint(bp))
```

Grossarl

Marriages and Births in Grossarl

Description

Data about the number of marriages and illegitimate and legitimate births in the Austrian Alpine village Grossarl in the 18th and 19th century.

Usage

```
data(Grossarl)
```

Format

Grossarl is a data frame containing 5 annual time series (1700 - 1899) and 3 factors coding policy interventions.

marriages time series. Number of marriages,

illegitimate time series. Number of illegitimate births,

legitimate time series. Number of legitimate births,

fraction time series. Fraction of illegitimate births,

lag.marriages time series. Number of marriages in the previous year,

politics factor coding 4 different political regimes,

morals factor coding 5 different moral regulations,

nuptiality factor coding 5 different marriage restrictions.

Details

The data frame contains historical demographic data from Grossarl, a village in the Alpine region of Salzburg, Austria, during the 18th and 19th century. During this period, the total population of Grossarl did not vary much on the whole, with the very exception of the period of the protestant emigrations in 1731/32.

Especially during the archbishopric, moral interventions aimed at lowering the proportion of illegitimate baptisms. For details see the references.

Source

Parish registers provide the basic demographic series of baptisms (which is almost equivalent to births in the study area) and marriages. For more information see Veichtlbauer et al. (2002).

References

Veichtlbauer O., Hanser E., Zeileis A., Leisch F. (2002), The Impact Of Policy Interventions on a Pre-Industrial Population System in the Austrian Alps, forthcoming.

Zeileis A., Veichtlbauer O. (2002), Policy Interventions Affecting Illegitimacy in Preindustrial Austria: A Structural Change Analysis, In R. Dutter (ed.), *Festschrift 50 Jahre Österreichische Statistische Gesellschaft*, 133-146, Österreichische Statistische Gesellschaft, <http://www.statistik.tuwien.ac.at/oezstat/>.

Examples

```
data(Grossarl)

## illegitimate births
#####
## lm + MOSUM
plot(Grossarl$fraction)
fm.min <- lm(fraction ~ politics, data = Grossarl)
fm.max <- lm(fraction ~ politics + morals + nuptiality + lag.marriages,
            data = Grossarl)
fm.final <- step(fm.max)
lines(ts(fitted(fm.min), start = 1700), col = 3)
lines(ts(fitted(fm.final), start = 1700), col = 4)
mos.min <- efp(fraction ~ politics, data = Grossarl, type = "OLS-MOSUM")
mos.final <- efp(fraction ~ politics + morals + nuptiality, data = Grossarl,
                type = "OLS-MOSUM")
plot(mos.min)
lines(mos.final, lty = 2)

## dating
bp <- breakpoints(fraction ~ 1, data = Grossarl, h = 0.1)
summary(bp)
## RSS, BIC, AIC
plot(bp)
plot(0:8, AIC(bp), type = "b")

## probably use 5 (or maybe 6) breakpoints and compare with
```

```

## coding of the factors as used by us
##
## politics                1803        1816 1850
## morals      1736 1753 1771 1803
## nuptiality                1803 1810 1816        1883
##
## m = 5                1753 1785                1821 1856 1878
## m = 6      1734 1754 1785                1821 1856 1878
##                6    2    5                1    4    3

fm.bp <- lm(fraction ~ breakfactor(breakpoints(bp, breaks = 6)),
           data = Grossarl)

plot(Grossarl$fraction)
lines(fitted(fm.final), col = 3)
lines(fitted(fm.bp), col = 4)

## marriages
#####
## lm + MOSUM
plot(Grossarl$marriages)
fm.min <- lm(marriages ~ politics, data = Grossarl)
fm.final <- lm(marriages ~ politics + morals + nuptiality, data = Grossarl)
lines(ts(fitted(fm.min), start = 1700), col = 3)
lines(ts(fitted(fm.final), start = 1700), col = 4)
mos.min <- efp(marriages ~ politics, data = Grossarl, type = "OLS-MOSUM")
mos.final <- efp(marriages ~ politics + morals + nuptiality, data = Grossarl,
                type = "OLS-MOSUM")

plot(mos.min)
lines(mos.final, lty = 2)

## dating
bp <- breakpoints(marriages ~ 1, data = Grossarl, h = 0.1)
summary(bp)
## RSS, BIC, AIC
plot(bp)
plot(0:8, AIC(bp), type = "b")

## probably use 3 (or maybe 4) breakpoints and compare with
## coding of the factors as used by us
##
## politics                1803        1816 1850
## morals      1736 1753 1771 1803
## nuptiality                1803 1810 1816        1883
##
## m = 3      1738                1813        1875
## m = 4      1738        1794                1814        1875
##                2          4                1          3

```

```
fm.bp <- lm(marriages ~ breakfactor(breakpoints(bp, breaks = 4)),
           data = Grossarl)

plot(Grossarl$marriages)
lines(fitted(fm.final), col = 3)
lines(fitted(fm.bp), col = 4)
```

| | |
|----------|--|
| USIncExp | <i>Income and Expenditures in the US</i> |
|----------|--|

Description

Personal income and personal consumption expenditures in the US between January 1959 and February 2001 (seasonally adjusted at annual rates).

Usage

```
data(USIncExp)
```

Format

A multivariate monthly time series from 1959(1) to 2001(2) with variables

income monthly personal income (in billion US dollars),

expenditure monthly personal consumption expenditures (in billion US Dollars).

Source

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

References

A. Zeileis, F. Leisch, K. Hornik, C. Kleiber (2002), *strucchange*: An R Package for Testing for Structural Change in Linear Regression Models. *Journal of Statistical Software* 7(2), 1–38.

Examples

```

## These example are presented in the vignette distributed with this
## package, the code was generated by Stangle("strucchange-intro.Rnw")

#####
### chunk number 1: data
#####
library(strucchange)
data(USIncExp)
plot(USIncExp, plot.type = "single", col = 1:2, ylab = "billion US$")
legend(1960, max(USIncExp), c("income", "expenditures"),
      lty = c(1,1), col = 1:2, bty = "n")

#####
### chunk number 2: subset
#####
library(strucchange)
data(USIncExp)
library(ts)
USIncExp2 <- window(USIncExp, start = c(1985,12))

#####
### chunk number 3: ecm-setup
#####
coint.res <- residuals(lm(expenditure ~ income, data = USIncExp2))
coint.res <- lag(ts(coint.res, start = c(1985,12), freq = 12), k = -1)
USIncExp2 <- cbind(USIncExp2, diff(USIncExp2), coint.res)
USIncExp2 <- window(USIncExp2, start = c(1986,1), end = c(2001,2))
colnames(USIncExp2) <- c("income", "expenditure", "diff.income",
                        "diff.expenditure", "coint.res")
ecm.model <- diff.expenditure ~ coint.res + diff.income

#####
### chunk number 4: ts-used
#####
plot(USIncExp2[,3:5], main = "")

#####
### chunk number 5: efp
#####
ocus <- efp(ecm.model, type="OLS-CUSUM", data=USIncExp2)
me <- efp(ecm.model, type="ME", data=USIncExp2, h=0.2)

#####
### chunk number 6: efp-boundary
#####
bound.ocus <- boundary(ocus, alpha=0.05)

```



```
#####  
### chunk number 7: OLS-CUSUM  
#####  
plot(ocus)  
  
#####  
### chunk number 8: efp-boundary2  
#####  
plot(ocus, boundary = FALSE)  
lines(bound.ocus, col = 4)  
lines(-bound.ocus, col = 4)  
  
#####  
### chunk number 9: ME-null  
#####  
plot(me, functional = NULL)  
  
#####  
### chunk number 10: efp-sctest  
#####  
sctest(ocus)  
  
#####  
### chunk number 11: efp-sctest2  
#####  
sctest(ecm.model, type="OLS-CUSUM", data=USIncExp2)  
  
#####  
### chunk number 12: Fstats  
#####  
fs <- Fstats(ecm.model, from = c(1990, 1), to = c(1999,6), data = USIncExp2)  
  
#####  
### chunk number 13: Fstats-plot  
#####  
plot(fs)  
  
#####  
### chunk number 14: pval-plot  
#####  
plot(fs, pval=TRUE)  
  
#####  
### chunk number 15: aveF-plot  
#####  
plot(fs, aveF=TRUE)  
  
#####  
### chunk number 16: Fstats-sctest
```

```
#####
sctest(fs, type="expF")

#####
### chunk number 17: Fstats-sctest2
#####
sctest(ecm.model, type = "expF", from = 49, to = 162, data = USIncExp2)

#####
### chunk number 18: mefp
#####
USIncExp3 <- window(USIncExp2, start = c(1986, 1), end = c(1989,12))
me.mefp <- mefp(ecm.model, type = "ME", data = USIncExp3, alpha = 0.05)

#####
### chunk number 19: monitor1
#####
USIncExp3 <- window(USIncExp2, start = c(1986, 1), end = c(1990,12))
me.mefp <- monitor(me.mefp)

#####
### chunk number 20: monitor2
#####
USIncExp3 <- window(USIncExp2, start = c(1986, 1))
me.mefp <- monitor(me.mefp)
me.mefp

#####
### chunk number 21: monitor-plot
#####
plot(me.mefp)

#####
### chunk number 22: mefp2
#####
USIncExp3 <- window(USIncExp2, start = c(1986, 1), end = c(1989,12))
me.efp <- efp(ecm.model, type = "ME", data = USIncExp3, h = 0.5)
me.mefp <- mefp(me.efp, alpha=0.05)

#####
### chunk number 23: monitor3
#####
USIncExp3 <- window(USIncExp2, start = c(1986, 1))
me.mefp <- monitor(me.mefp)

#####
### chunk number 24: monitor-plot2
#####
plot(me.mefp)
```

`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 pval is TRUE). |
| <code>...</code> | currently not used. |

Value

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

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

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`

See Also

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

| | |
|---------------------------|---|
| <code>boundary.efp</code> | <i>Boundary for Empirical Fluctuation Processes</i> |
|---------------------------|---|

Description

Computes boundary for an object of class "efp"

Usage

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

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 Brownian bridge type processes only). |
| <code>functional</code> | indicates which functional should be applied to the empirical fluctuation process. See also plot.efp . |
| <code>...</code> | currently not used. |

Value

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

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)
```

| | |
|----------------------------|---|
| <code>boundary.mefp</code> | <i>Boundary Function for Monitoring of Structural Changes</i> |
|----------------------------|---|

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

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")
```

| | |
|-------------------------|--|
| <code>breakdates</code> | <i>Breakdates Corresponding to Breakpoints</i> |
|-------------------------|--|

Description

A generic function for computing the breakdates corresponding to breakpoints (and their confidence intervals).

Usage

```
breakdates(obj, format.times = FALSE, ...)
```

Arguments

| | |
|---------------------------|---|
| <code>obj</code> | An object of class "breakpoints", "breakpointsfull" or their confidence intervals as returned by <code>confint</code> . |
| <code>format.times</code> | logical. If set to TRUE a vector of strings with the formatted breakdates. See details for more information. |
| <code>...</code> | currently not used. |

Details

Breakpoints are the number of observations that are the last in one segment and breakdates are the corresponding points on the underlying time scale. The breakdates can be formatted which enhances readability in particular for quarterly or monthly time series. For example the breakdate 2002.75 of a monthly time series will be formatted to "2002(10)".

Value

A vector or matrix containing the breakdates.

See Also

[breakpoints](#), `confint`

Examples

```
require(ts)

## Nile data with one breakpoint: the annual flows drop in 1898
## because the first Ashwan dam was built
data(Nile)
plot(Nile)

bp.nile <- breakpoints(Nile ~ 1)
summary(bp.nile)
plot(bp.nile)

## compute breakdates corresponding to the
## breakpoints of minimum BIC segmentation
breakdates(bp.nile)
```



```
## confidence intervals
ci.nile <- confint(bp.nile)
breakdates(ci.nile)
ci.nile

plot(Nile)
lines(ci.nile)
```

| | |
|-------------|---------------------------------------|
| breakfactor | <i>Factor Coding of Segmentations</i> |
|-------------|---------------------------------------|

Description

Generates a factor encoding the segmentation given by a set of break-points.

Usage

```
breakfactor(obj, breaks = NULL, labels = NULL, ...)
```

Arguments

| | |
|--------|--|
| obj | An object of class "breakpoints" or "breakpointsfull" respectively. |
| breaks | an integer specifying the number of breaks to extract (only if obj is of class "breakpointsfull"), by default the minimum BIC partition is used. |
| labels | a vector of labels for the returned factor, by default the segments are numbered starting from "segment1". |
| ... | further arguments passed to factor. |

Value

A factor encoding the segmentation.

See Also

[breakpoints](#)

Examples

```
require(ts)

## Nile data with one breakpoint: the annual flows drop in 1898
## because the first Ashwan dam was built
data(Nile)
plot(Nile)

## compute breakpoints
bp.nile <- breakpoints(Nile ~ 1)

## fit and visualize segmented and unsegmented model
fm0 <- lm(Nile ~ 1)
fm1 <- lm(Nile ~ breakfactor(bp.nile, breaks = 1))

lines(fitted(fm0), col = 3)
lines(fitted(fm1), col = 4)
lines(bp.nile, breaks = 1)
```

`breakpoints` *Dating Breaks*

Description

Computation of breakpoints in regression relationships. Given a number of breaks the function computes the optimal breakpoints.

Usage

```
breakpoints(formula, h = 0.15, breaks = NULL, tol = 1e-15,
            data = list(), ...)
breakpoints(obj, breaks = NULL, ...)
summary(object, breaks = NULL, sort = TRUE,
        format.times = NULL, ...)
lines(x, breaks = NULL, lty = 2, ...)
```

Arguments

| | |
|--|--|
| <code>formula</code> | a symbolic description for the model in which breakpoints will be estimated. |
| <code>h</code> | minimal segment size either given as fraction relative to the sample size or as an integer giving the minimal number of observations in each segment. |
| <code>breaks</code> | integer specifying the maximal number of breaks to be calculated. By default the maximal number allowed by <code>h</code> is used. |
| <code>tol</code> | tolerance when <code>solve</code> is used. |
| <code>data</code> | an optional data frame containing the variables in the model. By default the variables are taken from the environment which <code>breakpoints</code> is called from. |
| <code>...</code> | currently not used. |
| <code>obj</code> , <code>object</code> | an object of class "breakpointsfull". |
| <code>sort</code> | logical. If set to TRUE <code>summary</code> tries to match the breakpoints from partitions with different numbers of breaks. |
| <code>format.times</code> | logical. If set to TRUE a vector of strings with the formatted breakdates is printed. See breakdates for more information. |
| <code>x</code> | an object of class "breakpoints". |
| <code>lty</code> | line type. |

Details

All procedures in this package are concerned with testing or assessing deviations from stability in the classical linear regression model

$$y_i = x_i^\top \beta + u_i$$

In many applications it is reasonable to assume that there are m breakpoints, where the coefficients shift from one stable regression relationship to a different one. Thus, there are $m + 1$ segments in which the regression coefficients are constant, and the model can be rewritten as

$$y_i = x_i^\top \beta_j + u_i \quad (i = i_{j-1} + 1, \dots, i_j, \quad j = 1, \dots, m + 1)$$

where j denotes the segment index. In practice the breakpoints i_j are rarely given exogenously, but have to be estimated. `breakpoints` estimates these breakpoints by minimizing the residual sum of squares (RSS) of the equation above.

The foundation for estimating breaks in time series regression models was given by Bai (1994) and was extended to multiple breaks by Bai (1997ab) and Bai & Perron (1998). `breakpoints` implements the algorithm described in Bai & Perron (2003) for simultaneous estimation of multiple breakpoints. The distribution function used for the confidence intervals for the breakpoints is given in Bai (1997b). The ideas behind this implementation are described in Zeileis et al. (2003).

The algorithm for computing the optimal breakpoints given the number of breaks is based on a dynamic programming approach. The underlying idea is that of the Bellman principle. The main computational effort is to compute a triangular RSS matrix, which gives the residual sum of squares for a segment starting at observation i and ending at i' with $i < i'$.

Given a formula as the first argument, `breakpoints` computes an object of class "breakpointsfull" which inherits from "breakpoints". This contains in particular the triangular RSS matrix and functions to extract an optimal segmentation. A summary of this object will give the breakpoints (and associated) breakdates for all segmentations up to the maximal number of breaks together with the associated RSS and BIC. These will be plotted if `plot` is applied and thus visualize the minimum BIC estimator of the number of breakpoints. From an object of class "breakpointsfull" an arbitrary number of breaks (admissible by the minimum segment size `h`) can be extracted by another application of `breakpoints`, returning an object of class "breakpoints". This contains only the breakpoints for the specified number of breaks and some model properties (number of observations, regressors, time series properties and the associated RSS) but not the triangular RSS matrix and related extractor functions. The set of breakpoints which is associated by default with a "breakpointsfull" object is the minimum BIC partition.

Breakpoints are the number of observations that are the last in one segment, it is also possible to compute the corresponding breakdates which are the breakpoints on the underlying time scale. The breakdates can be formatted which enhances readability in particular for quarterly or monthly time series. For example the breakdate 2002.75

of a monthly time series will be formatted to "2002(10)". See [breakdates](#) for more details.

From a "breakpointsfull" object confidence intervals for the breakpoints can be computed using the method of `confint`. The breakdates corresponding to the breakpoints can again be computed by [breakdates](#). The breakpoints and their confidence intervals can be visualized by `lines`.

The log likelihood as well as some information criteria can be computed using the methods for the `logLik` and `AIC`. As for linear models the log likelihood is computed on a normal model and the degrees of freedom are the number of regression coefficients multiplied by the number of segments plus the number of estimated breakpoints plus 1 for the error variance. More details can be found on the help page of the method [logLik.breakpoints](#).

As the maximum of a sequence of F statistics is equivalent to the minimum OLS estimator of the breakpoint in a 2-segment partition it can be extracted by `breakpoints` from an object of class "Fstats" as computed by [Fstats](#). However, this cannot be used to extract a larger number of breakpoints.

For illustration see the commented examples below and Zeileis et al. (2003).

value

An object of class "breakpoints" is a list with the following elements:

breakpoints the breakpoints of the optimal partition with the number of breaks specified,

RSS the associated RSS,

nobs the number of observations,

nreg the number of regressors,

call the function call,

datatsp the time series properties `tsp` of the data, if any, `c(0, 1, nobs)` otherwise.

If applied to a formula as first argument, `breakpoints` returns an object of class "breakpointsfull" (which inherits from "breakpoints"), that contains some additional (or slightly different) elements such as:

breakpoints the breakpoints of the minimum BIC partition,

RSS a function which takes two arguments i, j and computes the residual sum of squares for a segment starting at observation i and ending at j by looking up the corresponding element in the triangular RSS matrix `RSS.triang`,

RSS.triang a list encoding the triangular RSS matrix.

References

Bai J. (1994), Least Squares Estimation of a Shift in Linear Processes, *Journal of Time Series Analysis*, **15**, 453-472.

Bai J. (1997a), Estimating Multiple Breaks One at a Time, *Econometric Theory*, **13**, 315-352.

Bai J. (1997b), Estimation of a Change Point in Multiple Regression Models, *Review of Economics and Statistics*, **79**, 551-563.

Bai J., Perron P. (1998), Estimating and Testing Linear Models With Multiple Structural Changes, *Econometrica*, **66**, 47-78.

Bai J., Perron P. (2003), Computation and Analysis of Multiple Structural Change Models, *Journal of Applied Econometrics*, **18**, 1-22.

Zeileis A., Kleiber C., Krämer W., Hornik K. (2003), Testing and Dating of Structural Changes in Practice, *Computational Statistics and Data Analysis*, forthcoming.

Examples

```
require(ts)

## Nile data with one breakpoint: the annual flows drop in 1898
## because the first Ashwan dam was built
data(Nile)
plot(Nile)

## F statistics indicate one breakpoint
fs.nile <- Fstats(Nile ~ 1)
plot(fs.nile)
breakpoints(fs.nile)
lines(breakpoints(fs.nile))

## or
bp.nile <- breakpoints(Nile ~ 1)
summary(bp.nile)
```

```

## the BIC also chooses one breakpoint
plot(bp.nile)
breakpoints(bp.nile)

## fit null hypothesis model and model with 1 breakpoint
fm0 <- lm(Nile ~ 1)
fm1 <- lm(Nile ~ breakfactor(bp.nile, breaks = 1))
plot(Nile)
lines(fitted(fm0), col = 3)
lines(fitted(fm1), col = 4)
lines(bp.nile)

## confidence interval
ci.nile <- confint(bp.nile)
ci.nile
lines(ci.nile)

## UK Seatbelt data: a SARIMA(1,0,0)(1,0,0)_12 model
## (fitted by OLS) is used and reveals (at least) two
## breakpoints - one in 1973 associated with the oil crisis and
## one in 1983 due to the introduction of compulsory
## wearing of seatbelts in the UK.
data(UKDriverDeaths)
seatbelt <- log10(UKDriverDeaths)
seatbelt <- cbind(seatbelt, lag(seatbelt, k = -1), lag(seatbelt, k = -12))
colnames(seatbelt) <- c("y", "ylag1", "ylag12")
seatbelt <- window(seatbelt, start = c(1970, 1), end = c(1984,12))
plot(seatbelt[, "y"], ylab = expression(log[10](casualties)))

## testing
re.seat <- efp(y ~ ylag1 + ylag12, data = seatbelt, type = "RE")
plot(re.seat)

## dating
bp.seat <- breakpoints(y ~ ylag1 + ylag12, data = seatbelt, h = 0.1)
summary(bp.seat)
lines(bp.seat, breaks = 2)

## minimum BIC partition
plot(bp.seat)
breakpoints(bp.seat)
## the BIC would choose 0 breakpoints although the RE and supF test
## clearly reject the hypothesis of structural stability. Bai &
## Perron (2003) report that the BIC has problems in dynamic regressions.
## due to the shape of the RE process of the F statistics choose two
## breakpoints and fit corresponding models
bp.seat2 <- breakpoints(bp.seat, breaks = 2)
fm0 <- lm(y ~ ylag1 + ylag12, data = seatbelt)

```

```

fm1 <- lm(y ~ breakfactor(bp.seat2)/(ylag1 + ylag12) - 1, data = seatbelt)

## plot
plot(seatbelt[,"y"], ylab = expression(log[10](casualties)))
time.seat <- as.vector(time(seatbelt))
lines(time.seat, fitted(fm0), col = 3)
lines(time.seat, fitted(fm1), col = 4)
lines(bp.seat2)

## confidence intervals
ci.seat2 <- confint(bp.seat, breaks = 2)
ci.seat2
lines(ci.seat2)

```

```
confint.breakpointsfull
```

Confidence Intervals for Breakpoints

Description

Computes confidence intervals for breakpoints.

Usage

```

confint(object, parm = NULL, level = 0.95,
        breaks = NULL, het.reg = TRUE, het.err = TRUE, ...)
lines(x, col = 2, angle = 90, length = 0.05,
      code = 3, at = NULL, breakpoints = TRUE, ...)

```

Arguments

| | |
|--------|---|
| object | an object of class "breakpointsfull" as computed by breakpoints from a formula. |
| parm | the same as breaks, only one of the two should be specified. |
| level | the confidence level required. |
| breaks | an integer specifying the number of breaks to be used. By default the breaks of the minimum BIC partition are used. |

| | |
|---------------------------------------|---|
| <code>het.reg</code> | logical. Should heterogenous regressors be assumed? If set to FALSE the distribution of the regressors is assumed to be homogenous over the segments. |
| <code>het.err</code> | logical. Should heterogenous errors be assumed? If set to FALSE the distribution of the errors is assumed to be homogenous over the segments. |
| <code>x</code> | an object of class "confint.breakpoints" as returned by <code>confint</code> . |
| <code>col, angle, length, code</code> | arguments passed to <code>arrows</code> . |
| <code>at</code> | position on the y axis, where the confidence arrows should be drawn. By default they are drawn at the bottom of the plot. |
| <code>breakpoints</code> | logical. If TRUE vertical lines for the breakpoints are drawn. |
| <code>...</code> | <i>currently not used.</i> |

Details

As the breakpoints are integers (observation numbers) the corresponding confidence intervals are also rounded to integers.

The distribution function used for the computation of confidence intervals of breakpoints is given in Bai (1997). The procedure, in particular the usage of heterogenous regressors and/or errors, is described in more detail in Bai & Perron (2003).

The breakpoints should be computed from a formula with `breakpoints`, then the confidence intervals for the breakpoints can be derived by `confint` and these can be visualized by `lines`. For an example see below.

Value

A matrix containing the breakpoints and their lower and upper confidence boundary for the given level.

References

Bai J. (1997), Estimation of a Change Point in Multiple Regression Models, *Review of Economics and Statistics*, **79**, 551-563.

Bai J., Perron P. (2003), Computation and Analysis of Multiple Structural Change Models, *Journal of Applied Econometrics*, **18**, 1-22.

See Also

[breakpoints](#)

Examples

```
require(ts)

## Nile data with one breakpoint: the annual flows drop in 1898
## because the first Ashwan dam was built
data(Nile)
plot(Nile)

## dating breaks
bp.nile <- breakpoints(Nile ~ 1)
ci.nile <- confint(bp.nile, breaks = 1)
lines(ci.nile)
```

| | |
|-------|---|
| covHC | <i>Heteroskedasticity-Consistent Covariance Matrix Estimation</i> |
|-------|---|

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 fitted.

| | |
|------|---|
| type | a character string specifying the estimation type. For details see below. |
| tol | tolerance when solve is used |
| data | an optional data frame containing the variables in the model. By default the variables are taken from the environment which covHC 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)))
```

| | |
|-------|------------------------------|
| durab | <i>US Labor Productivity</i> |
|-------|------------------------------|

Description

US labor productivity in the manufacturing/durables sector.

Usage

```
data(durab)
```

Format

`durab` is a multivariate monthly time series from 1947(3) to 2001(4) with variables

`y` growth rate of the Industrial Production Index to average weekly labor hours in the manufacturing/durables sector,

`lag` lag 1 of the series `y`,

Source

The data set is available from Bruce Hansen's homepage <http://www.ssc.wisc.edu/~bhansen/>. For more information see Hansen (2001).

References

Hansen B. (2001), The New Econometrics of Structural Change: Dating Breaks in U.S. Labor Productivity, *Journal of Economic Perspectives*, **15**, 117-128.

Zeileis A., Leisch F., Kleiber C., Hornik K. (2002), Monitoring Structural Change in Dynamic Econometric Models, Report 64, SFB "Adaptive Information Systems and Modelling in Economics and Management Science", Vienna University of Economics, <http://www.wu-wien.ac.at/am/reports.htm#78>.

Examples

```
data(durab)
## use AR(1) model as in Hansen (2001) and Zeileis et al. (2002)
durab.model <- y ~ lag

## historical tests
## OLS-based CUSUM process
ols <- efp(durab.model, data = durab, type = "OLS-CUSUM")
plot(ols)
## F statistics
fs <- Fstats(durab.model, data = durab, from = 0.1)
plot(fs)
## F statistics based on heteroskedasticity-consistent covariance matrix
fsHC <- Fstats(durab.model, data = durab, from = 0.1, cov.type = "HC")
plot(fsHC)

## monitoring
Durab <- window(durab, start=1964, end = c(1979, 12))
ols.efp <- efp(durab.model, type = "OLS-CUSUM", data = Durab)
newborder <- function(k) 1.5778*k/192
ols.mefp <- mefp(ols.efp, period=2)
ols.mefp2 <- mefp(ols.efp, border=newborder)
Durab <- window(durab, start=1964)
ols.mon <- monitor(ols.mefp)
ols.mon2 <- monitor(ols.mefp2)
plot(ols.mon)
lines(boundary(ols.mon2), col = 2)

## dating
bp <- breakpoints(durab.model, data = durab)
summary(bp)
plot(summary(bp))

plot(ols)
lines(breakpoints(bp, breaks = 1), col = 3)
lines(breakpoints(bp, breaks = 2), col = 4)
plot(fs)
lines(breakpoints(bp, breaks = 1), col = 3)
lines(breakpoints(bp, breaks = 2), col = 4)
```

| | |
|-----|--|
| efp | <i>Empirical Fluctuation Processes</i> |
|-----|--|

Description

Computes an empirical fluctuation process according to a specified method from the generalized fluctuation test framework, which includes CUSUM and MOSUM tests based on recursive or OLS residuals, parameter estimates or ML scores (OLS first order conditions).

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 efp 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 type is either "RE" or "ME") |
| tol | tolerance when solve 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 MOv-ing 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 `type` is either "RE" 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. The recursive estimates test is also called fluctuation test, therefore setting `type` to "fluctuation" was used to specify it in earlier versions of *strucchange*. It still can be used now, but will be forced to "RE".

If `type` is "Score-CUSUM" or "Score-MOSUM" a $k+1$ -dimensional process will be returned, one for each score of the regression coefficients and one for the scores of the variance. The process gives the decorrelated cumulative sums of the ML scores (in a gaussian model) or first order conditions respectively (in an OLS framework).

If there is a single structural change point t^* , the recursive CUSUM path starts to depart from its mean 0 at t^* . The Brownian bridge type paths will have their respective peaks around t^* . The Brownian bridge increments type paths should have a strong change at t^* .

The function `plot` has a method to plot the empirical fluctuation process; with `sctest` the corresponding test on structural change can be performed.

Value

`efp` returns a list of class "efp" with components including

| | |
|----------------------|---|
| <code>process</code> | the fitted empirical fluctuation process of class "ts" or "mts" respectively, |
| <code>type</code> | a string with the type of the process fitted, |
| <code>nreg</code> | the number of regressors, |
| <code>nobs</code> | the number of observations, |

par the bandwidth h used.

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.

Hansen B. (1992), Testing for Parameter Instability in Linear Models, *Journal of Policy Modeling*, **14**, 517-533.

Hjort N.L., Koning A. (2002), Tests for Constancy of Model Parameters Over Time, *Nonparametric Statistics*, **14**, 113-132.

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., Leisch F., Hornik K., Kleiber C. (2002), *strucchange*: An R Package for Testing for Structural Change in Linear Regression Models, *Journal of Statistical Software*, **7**(2), 1-38.

Zeileis A., Hornik K. (2003), Generalized M-Fluctuation Tests for Parameter Instability, Report 80, SFB "Adaptive Information Systems and Modelling in Economics and Management Science", Vienna University of Economics, <http://www.wu-wien.ac.at/am/reports.htm#80>.

See Also

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

Examples

```

require(ts)

## Nile data with one breakpoint: the annual flows drop in 1898
## because the first Ashwan dam was built
data(Nile)
plot(Nile)

## test the null hypothesis that the annual flow remains constant
## over the years
## compute OLS-based CUSUM process and plot
## with standard and alternative boundaries
ocus.nile <- efp(Nile ~ 1, type = "OLS-CUSUM")
plot(ocus.nile)
plot(ocus.nile, alpha = 0.01, alt.boundary = TRUE)
## calculate corresponding test statistic
sctest(ocus.nile)

## UK Seatbelt data: a SARIMA(1,0,0)(1,0,0)_12 model
## (fitted by OLS) is used and reveals (at least) two
## breakpoints - one in 1973 associated with the oil crisis and
## one in 1983 due to the introduction of compulsory
## wearing of seatbelts in the UK.
data(UKDriverDeaths)
seatbelt <- log10(UKDriverDeaths)
seatbelt <- cbind(seatbelt, lag(seatbelt, k = -1), lag(seatbelt, k = -12))
colnames(seatbelt) <- c("y", "ylag1", "ylag12")
seatbelt <- window(seatbelt, start = c(1970, 1), end = c(1984,12))
plot(seatbelt[, "y"], ylab = expression(log[10](casualties)))

## use RE process
re.seat <- efp(y ~ ylag1 + ylag12, data = seatbelt, type = "RE")
plot(re.seat)
plot(re.seat, functional = NULL)
sctest(re.seat)

```

```
logLik.breakpoints
```

Log Likelihood and Information Criteria for Breakpoints

Description

Computation of log likelihood and AIC type information criteria for partitions given by breakpoints.

Usage

```
logLik(object, breaks = NULL, ...)  
AIC(object, breaks = NULL, ..., k = 2)
```

Arguments

| | |
|---------------------|--|
| <code>object</code> | an object of class "breakpoints" or "breakpointsfull". |
| <code>breaks</code> | if <code>object</code> is of class "breakpointsfull" the number of breaks can be specified. |
| <code>...</code> | <i>currently not used.</i> |
| <code>k</code> | the penalty parameter to be used, the default $k = 2$ is the classical AIC, $k = \log(n)$ gives the BIC, if n is the number of observations. |

Details

As for linear models the log likelihood is computed on a normal model and the degrees of freedom are the number of regression coefficients multiplied by the number of segments plus the number of estimated breakpoints plus 1 for the error variance.

If AIC is applied to an object of class "breakpointsfull" breaks can be a vector of integers and the AIC for each corresponding partition will be returned. By default the maximal number of breaks stored in the object is used. See below for an example.

Value

An object of class "logLik" or a simple vector containing the AIC respectively.

See Also

[breakpoints](#)

Examples

```
require(ts)
```

```

## Nile data with one breakpoint: the annual flows drop in 1898
## because the first Ashwan dam was built
data(Nile)
plot(Nile)

bp.nile <- breakpoints(Nile ~ 1)
summary(bp.nile)
plot(bp.nile)

## BIC of partitions with 0 to 5 breakpoints
plot(0:5, AIC(bp.nile, k = log(bp.nile$nobs)), type = "b")
## AIC
plot(0:5, AIC(bp.nile), type = "b")

## BIC, AIC, log likelihood of a single partition
bp.nile1 <- breakpoints(bp.nile, breaks = 1)
AIC(bp.nile1, k = log(bp.nile1$nobs))
AIC(bp.nile1)
logLik(bp.nile1)

```

mefp

Monitoring of Empirical Fluctuation Processes

Description

Online monitoring of structural breaks in a linear regression model. A sequential fluctuation test based on parameter estimates or OLS residuals signals structural breaks.

Usage

```

mefp(obj, ...)

mefp(formula, type = c("OLS-CUSUM", "OLS-MOSUM", "RE", "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>efp</code> is called from. |
| <code>type</code> | specifies which type of fluctuation process will be computed. |
| <code>h</code> | (only used for MOSUM/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 "efp" (for <code>mefp</code>) or "mefp" (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 MOSUM/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 type 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 "RE". |
| <code>border</code> | An optional user-specified border function for the empirical process. <i>This argument is under development.</i> |

verbose If TRUE, signal breaks by text output.
... Currently not used.

Details

`mefp` creates an object of class "mefp" either from a model formula or from an object of class "efp". In addition to the arguments of `efp`, 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 "efp" 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.

The recursive estimates test is also called fluctuation test, therefore setting type to "fluctuation" was used to specify it in earlier versions of `strucchange`. It still can be used now, but will be forced to "RE"

References

Leisch F., Hornik K., Kuan C.-M. (2000), Monitoring Structural Changes with the Generalized Fluctuation Test, *Econometric Theory*, **16**, 835-854.

Zeileis A., Leisch F., Kleiber C., Hornik K. (2003), Monitoring Structural Change in Dynamic Econometric Models, Report 64, SFB "Adaptive Information Systems and Modelling in Economics and Management Science", Vienna University of Economics, <http://www.wu-wien.ac.at/am/reports.htm#64>.

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.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). |
| <code>alpha</code> | numeric from interval (0,1) indicating the confidence level for which the boundary of the supF test will be computed. |
| <code>boundary</code> | logical. If set to FALSE the boundary will be computed but not plotted. |
| <code>aveF</code> | 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. |
| <code>xlab, ylab, ylim, ...</code> | high-level plot function parameters. |

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")

```

plot.efp

Plot Empirical Fluctuation Process

Description

Plot and lines 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", ...)
lines(x, functional = "max", ...)

```

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. |

| | |
|------------------------------------|---|
| <code>alt.boundary</code> | logical. If set to TRUE alternative boundaries (instead of the standard linear boundaries) will be plotted (for CUSUM processes only). |
| <code>boundary</code> | logical. If set to FALSE the boundary will be computed but not plotted. |
| <code>functional</code> | indicates which functional should be applied to the process before plotting and which boundaries should be used. If set to NULL a multiple process with boundaries for the "max" functional is plotted. For more details see below. |
| <code>main, ylim, ylab, ...</code> | high-level plot function parameters. |

Details

Plots are available for the "max" functional for all process types. For Brownian bridge type processes the maximum or mean squared Euclidian norm ("maxL2" and "meanL2") can be used for aggregating before plotting. No plots are available for the "range" functional.

Alternative boundaries that are proportional to the standard deviation of the corresponding limiting process are available for processes with Brownian motion or Brownian bridge limiting 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 for structural change can be performed.

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.mefp *Plot Methods for mefp Objects*

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.

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)
```

 recresid

Recursive Residuals

Description

A generic function for computing the recursive residuals (standardized one step prediction errors) of a linear regression model.

Usage

```
recresid(x, y, ..., tol = 1e-7)
recresid(formula, data = list(), ..., tol = 1e-7)
recresid(x, data = list(), ..., tol = 1e-7)
```

Arguments

| | |
|----------------------------|---|
| <code>x, y, formula</code> | specification of the linear regression model: either by a regressor matrix <code>x</code> and a response variable <code>y</code> , or by a formula or by a fitted object <code>x</code> of class "lm". |
| <code>data</code> | an optional data frame containing the variables in the model. By default the variables are taken from the environment which <code>recresid</code> is called from. Specifying data might also be necessary when applying <code>recresid</code> to a fitted model of class "lm" if this does not contain the regressor matrix and the response. |
| <code>...</code> | <i>currently not used.</i> |
| <code>tol</code> | tolerance when solve is used. |

Details

Under the usual assumptions for the linear regression model the recursive residuals are (asymptotically) normal and i.i.d. (see Brown, Durbin, Evans (1975) for details).

Value

A vector containing the recursive residuals.

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.

See Also

[efp](#)

Examples

```
x <- rnorm(100)
x[51:100] <- x[51:100] + 2
rr <- recresid(x ~ 1)
plot(cumsum(rr), type = "l")

plot(efp(x ~ 1, type = "Rec-CUSUM"))
```

`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`

Examples

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

`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

`x` an object of class "Fstats".

`type` a character string specifying which test will be performed.

`asymptotic` logical. Only necessary if `x` contains just a single F statistic and `type` 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.

`...` 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.

Andrews D.W.K., Ploberger W. (1994), Optimal tests when a nuisance parameter is present only under the alternative, *Econometrica*, **62**, 1383-1414.

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",
        "maxL2", "meanL2"), from = 0.15, to = NULL, point = 0.5,
        asymptotic = FALSE, data, ...)

```

Arguments

| | |
|---------|--|
| formula | a formula describing the model to be tested. |
| type | a character string specifying the structural change test that is to be performed. Besides the tests types described in efp and sctest.Fstats . The Chow test and the Nyblom-Hansen test can be performed by setting type to "Chow" or "Nyblom-Hansen", respectively. |
| h | 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>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 used to aggregate the empirical fluctuation processes to a test statistic. |
| <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 $1 - \text{from}$. F statistics will be calculated for the observations $(n*\text{from}) : (n*\text{to})$, when n 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 $n - \text{from}$. (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 $\text{floor}(n*0.5)$ if n is the number of observations in the model. |
| <code>asymptotic</code> | logical. If TRUE 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 and the Nyblom-Hansen test are available explicitly here.

Value

an object of class "htest" containing:

| | |
|------------------------|---------------------------|
| <code>statistic</code> | the test statistic |
| <code>p.value</code> | the corresponding p value |

method a character string with the method used
 data.name a character string with the data name

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))
```

sctest.efp *Generalized Fluctuation Tests*

Description

Performs a generalized fluctuation test.

Usage

```
sctest(x, alt.boundary = FALSE,
       functional = c("max", "range", "maxL2", "meanL2"), ...)
```

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 empirical fluctuation process. |
| ... | 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. Similarly the critical values for the maximum and mean squared Euclidian norm ("maxL2" and "meanL2") are tabulated for confidence levels between 0.2 and 0.005.

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 |

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)
```

`strucchange.internal`

Internal strucchange objects

Description

These are not to be called by the user.

Bibliography

- Andrews, D. W. K. (1993), "Tests for Parameter Instability and Structural Change With Unknown Change Point," *Econometrica*, 61, 821–856.
- Andrews, D. W. K. and Ploberger, W. (1994), "Optimal Tests When a Nuisance Parameter is Present Only Under the Alternative," *Econometrica*, 62, 1383–1414.
- Bai, J. (1997a), "Estimating Multiple Breaks One at a Time," *Econometric Theory*, 13, 315–352.
- (1997b), "Estimation of a Change Point in Multiple Regression Models," *Review of Economics and Statistics*, 79, 551–563.
- Bai, J. and Perron, P. (1998), "Estimating and Testing Linear Models With Multiple Structural Changes," *Econometrica*, 66, 47–78.
- (2003), "Computation and Analysis of Multiple Structural Change Models," *Journal of Applied Econometrics*, 18, 1–22.
- Balke, N. S. (1993), "Detecting Level Shifts in Time Series," *Journal of Business and Economic Statistics*, 11, 81–92.
- Basseville, M. and Benveniste, A. (1986), *Detection of Abrupt Changes in Sig-*

- nals and Dynamical Systems*, vol. 77 of *Lecture Notes in Control and Information Sciences*, New York: Springer.
- Basseville, M. and Nikiforov, I. V. (1993), *Detection of Abrupt Changes: Theory and Application*, Englewood Cliffs, NJ: Prentice-Hall.
- Bauer, P. and Hackl, P. (1978), "The Use of MOSUMS for Quality Control," *Technometrics*, 20, 431–436.
- Becker, R. A., Chambers, J. M., and Wilks, A. R. (1988), *The New S Language*, London: Chapman & Hall.
- Bera, A. K. and Bilias, Y. (2002), "The MM, ME, ML, EL, EF and GMM Approaches to Estimation: A Synthesis," *Journal of Econometrics*, 107, 51–86.
- Billingsley, P. (1999), *Convergence of Probability Measures*, New York: Wiley, 2nd ed.
- Breiman, L. (2001), "Statistical Modeling: The Two Cultures," *Statistical Science*, 16, 199–231.
- Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984), *Classification and Regression Trees*, Belmont: Wadsworth.
- Brown, R. L., Durbin, J., and Evans, J. M. (1975), "Techniques for Testing the Constancy of Regression Relationships over Time," *Journal of the Royal Statistical Society B*, 37, 149–163.
- Carsoule, F. and Franses, P. H. (2003), "A Note on Monitoring Time-Varying Parameters in an Autoregression," *Metrika*, 57, 51–62.

- Chambers, J. M. (1998), *Programming with Data*, New York: Springer, ISBN 0-387-98503-4.
- Chambers, J. M. and Hastie, T. J. (1992), *Statistical Models in S*, London: Chapman & Hall.
- Chong, T. T.-L. (1995), "Partial Parameter Consistency in a Misspecified Structural Change Model," *Economics Letters*, 49, 351–357.
- Chow, G. C. (1960), "Tests of Equality Between Sets of Coefficients in Two Linear Regressions," *Econometrica*, 28, 591–605.
- Chu, C.-S. J., Hornik, K., and Kuan, C.-M. (1995a), "MOSUM Tests for Parameter Constancy," *Biometrika*, 82, 603–617.
- (1995b), "The Moving-Estimates Test for Parameter Stability," *Econometric Theory*, 11, 669–720.
- Chu, C.-S. J., Stinchcombe, M., and White, H. (1996), "Monitoring Structural Change," *Econometrica*, 64, 1045–1065.
- Cleveland, W. S. (1993), *Visualizing Data*, Summit, New Jersey: Hobart Press.
- Cobb, G. W. (1978), "The Problem of the Nile: Conditional Solution to a Change-Point Problem," *Biometrika*, 65, 243–251.
- Cooper, S. J., Piehl, A. M., Braga, A. A., and Kennedy, D. M. (2001), "Testing for Structural Breaks in the Evaluation of Programs," Unpublished Paper, John F. Kennedy School of Government.
- Cribari-Neto, F. and Zarkos, S. G. (1999), "R: Yet Another Econometric Programming Environment," *Journal of Applied Econometrics*, 14, 319–329.

- Dümbgen, L. (1991), "The Asymptotic Behavior of Some Nonparametric Change-Point Estimators," *Annals of Statistics*, 19, 1471–1495.
- Godambe, V. P. (1960), "An Optimum Property of Regular Maximum Likelihood Estimation," *Annals of Mathematical Statistics*, 31, 1208–1211.
- (1985), "The Foundations of Finite-Sample Inference in Stochastic Processes," *Biometrika*, 48, 419–428.
- Hansen, B. E. (1992), "Testing for Parameter Instability in Linear Models," *Journal of Policy Modeling*, 14, 517–533.
- (1997), "Approximate Asymptotic p Values for Structural-Change Tests," *Journal of Business & Economic Statistics*, 15, 60–67.
- (2001), "The New Econometrics of Structural Change: Dating Breaks in U.S. Labor Productivity," *Journal of Economic Perspectives*, 15, 117–128.
- Hansen, L. (1982), "Large Sample Properties of Generalized Method of Moments Estimators," *Econometrica*, 50, 1029–1054.
- Harvey, A. C. and Durbin, J. (1986), "The Effects of Seat Belt Legislation on British Road Casualties: A Case Study in Structural Time Series Modelling (with Discussion)," *Journal of the Royal Statistical Society A*, 149, 187–227.
- Hawkins, D. M. (1976), "Point Estimation of the Parameters of a Piecewise Regression Model," *Applied Statistics*, 25, 51–57.
- (2001), "Fitting Multiple Change-Point Models to Data," *Computational Statistics & Data Analysis*, 37, 323–341.

- Hjort, N. L. and Koning, A. (2002), "Tests for Constancy of Model Parameters Over Time," *Nonparametric Statistics*, 14, 113–132.
- Hornik, K. and Leisch, F. (eds.) (2002), *R News*, <http://CRAN.R-project.org/doc/Rnews/>, ISSN 1609-3631.
- Hornik, K., Leisch, F., and Zeileis, A. (eds.) (2003), *Proceedings of the 3rd International Workshop on Distributed Statistical Computing (DSC 2003)*, Technische Universität Wien, Vienna, Austria, ISSN 1609-395X.
- Hothorn, T. and Lausen, B. (2003), "On the Exact Distribution of Maximally Selected Rank Statistics," *Computational Statistics & Data Analysis*, 43, 121–137.
- Huber, P. J. (1964), "Robust Estimation of a Location Parameter," *Annals of Mathematical Statistics*, 35, 73–101.
- (1972), "Robust Statistics: A Review," *Annals of Mathematical Statistics*, 43, 1041–1067.
- Ihaka, R. and Gentleman, R. (1996), "R: A Language for Data Analysis and Graphics," *Journal of Computational and Graphical Statistics*, 5, 299–314.
- Kennedy, D. M., Piehl, A. M., and Braga, A. A. (1996), "Youth Violence in Boston: Gun Markets, Serious Youth Offenders, and a Use-Reduction Strategy," *Law and Contemporary Problems*, 59, 147–183.
- Krämer, W., Ploberger, W., and Alt, R. (1988), "Testing for Structural Change in Dynamic Models," *Econometrica*, 56, 1355–1369.
- Krämer, W. and Schotman, P. (1992), "Range vs. Maximum in the OLS-based Version of the CUSUM Test," *Economics Letters*, 40, 379–381.

- Krämer, W. and Sonnberger, H. (1986), *The Linear Regression Model Under Test*, Heidelberg: Physica-Verlag.
- Kuan, C.-M. and Chen, M.-Y. (1994), "Implementing the Fluctuation and Moving-Estimates Tests in Dynamic Econometric Models," *Economics Letters*, 44, 235–239.
- Kuan, C.-M. and Hornik, K. (1995), "The Generalized Fluctuation Test: A Unifying View," *Econometric Reviews*, 14, 135–161.
- Lausen, B. and Schumacher, M. (1992), "Maximally Selected Rank Statistics," *Biometrics*, 48, 73–85.
- Leisch, F. (2002), "Sweave: Dynamic Generation of Statistical Reports Using Literate Data Analysis," in *Compstat 2002 — Proceedings in Computational Statistics*, eds. Härdle, W. and Rönz, B., Physika Verlag, Heidelberg, Germany, pp. 575–580, ISBN 3-7908-1517-9.
- Leisch, F., Hornik, K., and Kuan, C.-M. (2000), "Monitoring Structural Changes With the Generalized Fluctuation Test," *Econometric Theory*, 16, 835–854.
- Liang, K.-Y. and Zeger, S. L. (1986), "Longitudinal Data Analysis Using Generalized Linear Models," *Biometrika*, 73, 13–22.
- Lumley, T. and Heagerty, P. (1999), "Weighted Empirical Adaptive Variance Estimators for Correlated Data Regression," *Journal of the Royal Statistical Society B*, 61, 459–477.
- Lütkepohl, H., Teräsvirta, T., and Wolters, J. (1999), "Investigating Stability and Linearity of a German M1 Money Demand Function," *Journal of Applied Econometrics*, 14, 511–525.

- Mazanec, J. A. and Strasser, H. (2000), *A Nonparametric Approach to Perceptions-Based Market Segmentation: Foundations*, Berlin: Springer.
- McCullagh, P. and Nelder, J. A. (1989), *Generalized Linear Models*, London: Chapman & Hall, 2nd ed.
- Meyer, D., Zeileis, A., and Hornik, K. (2003), "Visualizing Independence Using Extended Association Plots," in *Proceedings of the 3rd International Workshop on Distributed Statistical Computing, Vienna, Austria*, eds. Hornik, K., Leisch, F., and Zeileis, A., ISSN 1609-395X.
- Nyblom, J. (1989), "Testing for the Constancy of Parameters Over Time," *Journal of the American Statistical Association*, 84, 223–230.
- Pastor-Barriuso, R., Guallar, E., and Coresh, J. (2003), "Transition Models for Change-Point Estimation in Logistic Regression," *Statistics in Medicine*, 22, 1141–1162.
- Ploberger, W. and Krämer, W. (1992), "The CUSUM Test With OLS Residuals," *Econometrica*, 60, 271–285.
- (1996), "A Trend-Resistant Test for Structural Change Based on OLS Residuals," *Journal of Econometrics*, 70, 175–185.
- Ploberger, W., Krämer, W., and Kontrus, K. (1989), "A New Test for Structural Stability in the Linear Regression Model," *Journal of Econometrics*, 40, 307–318.
- Racine, J. and Hyndman, R. (2002), "Using R To Teach Econometrics," *Journal of Applied Econometrics*, 17, 175–189.
- Sargan, J. D. (1958), "The Estimation of Economic Relationships Using Instrumental Variables," *Econometrica*, 26, 393–415.

- Sen, P. K. (1984), "Recursive M-Tests for the Constancy of Multivariate Regression Relationships Over Time," *Sequential Analysis*, 3, 191–211.
- Sibbertsen, P. (2000), "Robust CUSUM-M Test in the Presence of Long-Memory Disturbances," Technical Report 28, SFB 475 "Reduction of Complexity for Multivariate Data Structures".
- Siegmund, D. (1985), *Sequential Analysis*, New York: Springer.
- Stasinopoulos, D. M. and Rigby, R. A. (1992), "Detecting Break Points in Generalised Linear Models," *Computational Statistics & Data Analysis*, 13, 461–471.
- Stefanski, L. A. and Boos, D. D. (2002), "The Calculus of M-Estimation," *American Statistician*, 56, 29–38.
- Sullivan, J. H. (2002), "Estimating the Locations of Multiple Change Points in the Mean," *Computational Statistics*, 17, 289–296.
- Tukey, J. W. (1962), "The Future of Data Analysis," *Annals of Mathematical Statistics*, 33, 1–67.
- Veichtlbauer, O., Hanser, E., Zeileis, A., and Leisch, F. (2002), "The Impact of Policy Interventions on a Pre-industrial Population System in the Austrian Alps," Unpublished Manuscript.
- White, H. (1994), *Estimation, Inference and Specification Analysis*, Cambridge University Press.
- Wilkinson, G. N. and Rogers, C. E. (1973), "Symbolic Description of Factorial Models for Analysis of Variance," *Applied Statistics*, 22, 392–399.

- Zeileis, A. (2000), "*p*-Werte und alternative Schranken von CUSUM-Tests," Master's thesis, Fachbereich Statistik, Universität Dortmund, in German.
- (2001), "strucchange: Testing for Structural Change in Linear Regression Relationships," *R News*, 1, 8–11.
- (2004), "Alternative Boundaries for CUSUM Tests," *Statistical Papers*, forthcoming.
- Zeileis, A. and Hornik, K. (2003), "Generalized M-Fluctuation Tests for Parameter Instability," Report 80, SFB "Adaptive Information Systems and Modelling in Economics and Management Science".
- Zeileis, A. and Hothorn, T. (2002), "Diagnostic Checking in Regression Relationships," *R News*, 2, 7–10.
- Zeileis, A., Kleiber, C., Krämer, W., and Hornik, K. (2003), "Testing and Dating of Structural Changes in Practice," *Computational Statistics and Data Analysis*, 44, 109–123.
- Zeileis, A., Leisch, F., Hornik, K., and Kleiber, C. (2002), "strucchange: An R Package for Testing for Structural Change in Linear Regression Models," *Journal of Statistical Software*, 7, 1–38.
- Zeileis, A., Leisch, F., Kleiber, C., and Hornik, K. (2004), "Monitoring Structural Change in Dynamic Econometric Models," *Journal of Applied Econometrics*, forthcoming.
- Zeileis, A. and Veichtlbauer, O. (2002), "Policy Interventions Affecting Illegitimacy in Preindustrial Austria: A Structural Change Analysis," in

Festschrift 50 Jahre Österreichische Statistische Gesellschaft, ed. Dutter, R.,
Österreichische Statistische Gesellschaft, pp. 133–146, ISSN 1026-597X.

List of Figures

| | | |
|-----|--|----|
| 1.1 | Annual flows of the river Nile before and after opening of the first Aswan dam | 6 |
| 2.1 | Score-based fluctuation process (mean L_2 norm) for German M1 data | 41 |
| 2.2 | Illegitimate births in Großarl and the binomial CUSUM process | 42 |
| 2.3 | Artificial binary data | 44 |
| 2.4 | 2-dimensional binomial CUSUM process | 45 |
| 2.5 | The Boston homicide data and the Poisson CUSUM process | 47 |
| 3.1 | Personal income and personal consumption expenditures in the US | 54 |
| 3.2 | Time series used – first differences and cointegration residuals | 56 |
| 3.3 | OLS-based CUSUM process | 65 |
| 3.4 | 3-dimensional ME process | 66 |
| 3.5 | OLS-based CUSUM process with mean L_2 norm | 67 |
| 3.6 | F statistics | 72 |
| 3.7 | Monitoring structural change with bandwidth $h = 1$ | 78 |

| | | |
|------|---|-----|
| 3.8 | Monitoring structural change with bandwidth $h = 0.5$ | 78 |
| 4.1 | OLS-based CUSUM process for the Nile data | 89 |
| 4.2 | F statistics for the Nile data | 90 |
| 4.3 | BIC for models with m breakpoints | 91 |
| 4.4 | Fitted models for the Nile data | 93 |
| 4.5 | The seatbelt data | 94 |
| 4.6 | Structural change tests for the seatbelt data | 95 |
| 4.7 | BIC for models with m breakpoints | 96 |
| 4.8 | Fitted model for the seatbelt data | 97 |
| 4.9 | The oil price data | 98 |
| 4.10 | BIC for models with m breakpoints | 99 |
| 4.11 | OLS-based CUSUM process for the oil price data | 99 |
| 4.12 | Fitted model for the oil price data | 100 |