

Validating multiple structural change models – A case study

Achim Zeileis^{a*}

Christian Kleiber^b

^a *Institut für Statistik & Mathematik, Wirtschaftsuniversität Wien, Austria*

^b *Institut für Wirtschafts- und Sozialstatistik, Universität Dortmund, Germany*

SUMMARY

In a recent article, [Bai and Perron \(2003, *Journal of Applied Econometrics*\)](#) present a comprehensive discussion of computational aspects of multiple structural change models along with several empirical examples. Here, we report on the results of a replication study using the R statistical software package. We are able to verify most of their findings; however, some confidence intervals associated with breakpoints cannot be reproduced. These confidence intervals require computation of the quantiles of a nonstandard distribution, the distribution of the argmax functional of a certain stochastic process. Interestingly, the difficulties appear to be due to numerical problems in `GAUSS`, the software package used by Bai and Perron.

1. INTRODUCTION

Time series with multiple structural changes have recently attracted considerable attention in theoretical and applied econometric literature. [Bai \(1997\)](#) and [Bai and Perron \(1998\)](#) present asymptotic theory for inference on multiple breakpoints, in a companion paper ([Bai and Perron, 2003](#)) they provide a comprehensive and detailed discussion of computational aspects as well as several empirical examples.

The present paper aims at replicating the results from [Bai and Perron \(2003\)](#) using the R statistical software package ([R Development Core Team, 2004](#), see <http://www.R-project.org/>). Specifically, we employ the R package `strucchange` ([Zeileis, Leisch, Hornik and Kleiber, 2002](#)), which implements a large collection of methods for the analysis of structural change.

2. REPLICATION

The model considered here is the multiple linear regression model with m breaks (or, equivalently, $m + 1$ regimes)

$$y_t = x_t^\top \delta_j + u_t, \quad t = T_{j-1} + 1, \dots, T_j, \quad j = 1, \dots, m + 1, \quad (1)$$

where $T_0 = 0$ and $T_{m+1} = T$ (= sample size) by convention. The goal of the analysis is to determine the number m and locations T_j ($j = 1, \dots, m$) of the breakpoints. In practice, a search over m is conducted, for $m \leq m^*$, where m^* is fixed by the researcher. More importantly, a further parameter is at the researcher's disposal, the minimum number of observations per segment, h . Bai and Perron consider a slightly more general setup allowing for partial structural changes, in which only some of the regression coefficients are subject to shifts. The current version of `strucchange` does not support partial structural change models. However, this additional feature is not required

*Correspondence to: Achim Zeileis, Institut für Statistik & Mathematik, Wirtschaftsuniversität Wien, Augasse 2-6, A-1090 Wien, Austria. E-mail: Achim.Zeileis@wu-wien.ac.at

for two of the three Bai and Perron examples; for the third, estimates of the breakpoints—the parameters of main interest—remain unaffected.

Due to space restrictions, we confine ourselves to the first Bai and Perron example. In an extended version of the present paper (Zeileis and Kleiber, 2004), we provide further details on this and the remaining two data sets. For an AR(1) model of the UK inflation rate, most results are confirmed if the same bandwidth h is used. However, small changes in h lead to a competing model with different results. For a Phillips curve equation, our results are consistent with those of Bai and Perron.

In the following, we discuss the US ex-post real interest rate series for the period 1961(1)–1986(3), defined as the three-month treasury bill rate deflated by the CPI inflation rate. It is modelled by only a constant as regressor, utilizing a minimum segment size of $h = 15$ observations. This amounts to allowing for up to $m^* = 5$ breaks. Bai and Perron discuss this series in considerable detail, and their script for the analysis of this data set, **break.prg**, is available from the JAE data archive along with **brcode.src**, a collection of GAUSS functions upon which **break.prg** relies. This greatly facilitates our task. Following Bai and Perron, we consider a three-break model below.

The coefficient estimates (with standard errors in parentheses) and the corresponding confidence intervals at the 95% level are given in Table I. The standard errors are estimated utilizing a kernel HAC estimator with a quadratic spectral kernel, prewhitening using a VAR(1) model and an AR(1) approximation for the automatic bandwidth selection. The estimated breakpoints, 1966(4), 1972(3) and 1980(3), as well as the regression coefficients agree with those obtained by Bai and Perron, there are minor differences regarding the estimated standard errors. Overall, this part of the replication must be considered a success. However, there are considerable differences in the confidence intervals corresponding to the break dates, which cannot be explained by rounding errors in different computing environments. Specifically, the second confidence interval (1969(1)–1972(4)) extends much further to the left than the one reported by Bai and Perron (1970(3)–1972(4)). With some effort, we were able to track down the source of the problem, a so-called underflow for which there does not appear to be an easy workaround in GAUSS.

Table I: US *ex post* real interest rate (1961(1)–1986(3))

Coefficient estimates with three breaks ($h = 15$)			
$\hat{\delta}_1$	$\hat{\delta}_2$	$\hat{\delta}_3$	$\hat{\delta}_4$
1.82	0.87	-1.80	5.64
(0.19)	(0.15)	(0.50)	(0.59)
Corresponding breakpoint estimates			
\hat{T}_1	\hat{T}_2	\hat{T}_3	
1966(4)	1972(3)	1980(3)	
1965(2)–1969(3)	1969(1)–1972(4)	1980(1)–1981(1)	

The confidence intervals are derived from the distribution of the argmax functional of a process composed of two independent Brownian motions with different linear drifts and scales; see Bai (1997) for further details on this nonstandard distribution. This cumulative distribution function (CDF) depends on three parameters which are here associated with ratios of quadratic forms in the magnitude of the shifts and weighting matrices defined as segment-wise covariance matrices.

The CDF contains terms of the type $\exp(ax) \cdot \Phi(-b\sqrt{x})$, where Φ denotes the CDF of the standard normal distribution, a and b are functions of the residual standard errors pertaining to adjacent segments. For the confidence interval corresponding to the second breakpoint, a is approximately equal to 8.314 while b is approximately equal to 4.08, and the term must be evaluated for $x \in [-10, 300]$ (approximately). Numerically, for x in the vicinity of 300, the term $\exp(ax) \cdot \Phi(-b\sqrt{x})$ is the product of a rather large, $\exp(ax)$, and a rather small, $\Phi(-b\sqrt{x})$, number and it depends on the implementation as well as the routines used by the software package what the latter returns. We looked into this problem utilizing GAUSS v3.2.38, a good proxy for GAUSS v3.2.32, the version used by Bai and Perron.¹ Using their program **brcode.src**, GAUSS v3.2.38 returns values of the

¹We thank Pierre Perron for providing this information.

CDF up to approximately 0.95 (in fact, only values up to 0.9 are reliable—for larger values the computations cannot yield correct results on a PC as we shall see below), proceeds with some values in excess of 1 (!), and returns infinitely large function values thereafter. This results in a spurious kink in the CDF and in a bisection search (employed for computing the required quantile) stopping near this kink, yielding a 97.5% quantile in the vicinity of 86 and implying that the confidence interval reported by Bai and Perron is not valid. Our R package **strucchange** does not produce this spurious kink and returns a 97.5% quantile of 186.45. What is going on here? At first, this looks like a small programming error that is easy to correct: coding $\exp(ax) \cdot \Phi(-b\sqrt{x})$ term by term is not a numerically efficient way to handle this quantity. For large x , $\Phi(-4.08\sqrt{x})$ is tiny; in fact, for $x \geq 85$ it is smaller than the smallest positive number representable on a PC with 32-bit arithmetic. Columns 2 and 3 of Table II show how GAUSS and R deal with this underflow: R rounds the probability to the nearest number it can represent—as computers have to do for any real number—which in this case is 0. GAUSS, on the other hand, argues (as documented on the manual page for `cdfn`) that the CDF of the standard normal distribution is always greater than 0 and returns $2.225 \cdot 10^{-308}$, the smallest positive number it can work with. Evidently, neither approximation of the true probability is satisfactory in this situation and the term $\exp(ax) \cdot \Phi(-b\sqrt{x})$ must not be computed via $\Phi(x)$ for large x .

A natural solution would be to code the offending term in the form $\exp(ax + \log(\Phi(-b\sqrt{x})))$. Many software packages for statistics and econometrics, including GAUSS and R, provide a separate function for computing normal log-probabilities, $\log(\Phi(\cdot))$, directly. This is used in the implementation of the CDF in **strucchange** which allows us to compute the required 97.5% quantile. (Of course, we do not know what the true quantile is, since there is no benchmark for this nonstandard distribution. All we can say is that we have no reason to doubt that R returns a reasonable approximation to the truth.) However, in GAUSS the problem turns out to be more serious. While the spurious kink in the CDF disappears, GAUSS v3.2.38 is unable to go beyond the 90.7% quantile (it returns missing values thereafter). Clearly, for two-sided confidence intervals corresponding to conventional confidence levels, at least the 95% and preferably also the 97.5% and 99.5% quantiles are required. Columns 4 and 5 of Table III exhibit what GAUSS and R return for $\log(\Phi(-4.08\sqrt{x}))$, $x \in [80, 90]$, utilizing the GAUSS function `lncdfn` and its R counterpart `pnorm` with option `log.p = TRUE`. Evidently, `lncdfn` does not improve much upon `cdfn`: despite the log-probabilities having numerically non-critical values around -750, `lncdfn` returns $-\infty$ for $x \geq 89$. In contrast, R returns finite values for much larger arguments. Lacking access to the internals of GAUSS, we can only speculate that either an inaccurate expansion for $\log(\Phi)$ or a coding error is responsible for these results. A look at the documentation and the code of R reveals that `pnorm` utilizes an approximation to Φ outlined by Cody (1969), while $\log(\Phi)$ is handled via an asymptotic expansion (Abramowitz and Stegun, 1964, formula 26.2.13). In order to verify our findings we also asked a few colleagues to run our GAUSS programs in more recent versions of GAUSS, it emerges that the problems persist in GAUSS v5.0.22, v5.0.25, and v6.0.8.

Table II: Computation of $\Phi(y)$ and $\log(\Phi(y))$ with $y = -4.08\sqrt{x}$ in GAUSS and R

x	GAUSS cdfn(y)	R pnorm(y)	GAUSS lncdfn(y)	R pnorm(y, log.p = TRUE)
84.0	2.462e-306	2.462e-306	-703.690	-703.690
85.0	2.225e-308	0	-712.019	-712.019
86.0	2.225e-308	0	-720.348	-720.348
87.0	2.225e-308	0	-728.677	-728.677
88.0	2.225e-308	0	-737.006	-737.006
89.0	2.225e-308	0	$-\infty$	-745.335

Lacking an accurate function for evaluation of $\log(\Phi)$, GAUSS users may nonetheless obtain an approximate solution. We suggest using the elementary approximation (Feller, 1968, p. 175)

$$\Phi(-x) = 1 - \Phi(x) \approx \frac{1}{x\sqrt{2\pi}} \exp\left\{-\frac{x^2}{2}\right\} \quad (x > 0). \quad (2)$$

(In fact, (2) represents an upper bound that is fairly tight in the tails.) Utilizing $\log(\Phi(-x)) \approx -x^2/2 - \log(\sqrt{2\pi}) - \log(x)$, $x > 0$, in **brcode.src**, we find $q_{0.975} = 183.95$ for the distribution required in connection with the second breakpoint. This agrees with the value obtained in R based on more refined methods. (The difference to the quantile 186.45 reported above is due to slightly different HAC estimates of residual variances in GAUSS and R and not to the bound (2).)

In practical terms, using this quantile for computation of the confidence intervals it emerges that the intervals for the first and second break overlap (by three periods, see Table I), hence there appears to be considerable uncertainty as to the location of the first break.

Apart from the numerical problem mentioned above, there are further small differences between our results and those of Bai and Perron. Specifically, the lower limits of the remaining two confidence intervals differ by one observation from the results obtained utilizing **brcode.src**. As noted by Bai and Perron, the confidence intervals should be integer-valued like the breakpoints themselves. Therefore, instead of the interval $[a, b]$ computed from the asymptotic distribution the interval $[floor(a), ceiling(b)]$ should be returned. However, **brcode.src** computes the lower limit as $floor(a) - 1$ which is responsible for the difference of results. After shifting the lower limit to the right by one observation, R and GAUSS agree.

3. CONCLUSION

When the JAE started its Replication Section in early 2003, in the same issue in which the Bai and Perron paper was published, we felt that this paper provided an ideal opportunity to try out our R package **strucchange** on other authors' data. However, the task turned out to be rather difficult when we attempted to validate the confidence intervals. With some effort, we were able to trace these problems to an underflow problem that is not easily overcome in GAUSS. Lessons are threefold:

Evidently, numerical accuracy is important. Naturally, problems of this type are bound to occur in all software environments, and only further replication studies will bring them to the attention of researchers and software developers. When this paper was under review, P. Perron informed us that an improved version of **brcode.src** is now available from his Web page. In addition, Aptech, the makers of GAUSS, announced a fix of **lncdfn**.

We second [McCullough and Vinod \(2003\)](#) in suggesting journal archives with mandatory data and code. At present, the JAE only requires data, more often than not the corresponding code is not available. We are therefore grateful to Professors Bai and Perron for making their code publicly available without being obliged to. Without their code, this replication project would have had to stop halfway: GAUSS returns this, R returns that, and it would have been much harder to determine the source of the observed differences. With their code, we can confidently say that numerical problems in GAUSS are responsible for most of the differences.

Finally, as pointed out by [McCullough and Vinod \(2003\)](#), even data and code together will not be sufficient. In addition, anybody trying to replicate the results of some paper will require information on the software package including the version, the operating system and preferably the exact function calls to the package used in the analysis. Fortunately, there is a convenient way to achieve this (and more), at least for the combination of computing environment and text processor used by the authors of this paper, R and L^AT_EX. R provides the function **Sweave** (in the **utils** package) which is able to perform computations on integrated text documents that mix code (in R) and corresponding documentation (in L^AT_EX). See [Leisch and Rossini \(2003\)](#) for its use in connection with reviewing and replicating statistical research.

Computational details

Our results were obtained using R 2.1.1—with the packages **strucchange** 1.2–10, **sandwich** 1.0–1 and **zoo** 1.0–0—and were identical on various platforms including PCs running Debian GNU/Linux (with a 2.6.6 kernel) and Windows XP Professional, version 2002. GAUSS v3.2.38 was run under

Windows XP Professional, version 2002, on a 1.67 GHz AMD Athlon XP 2000+ processor. The full R and GAUSS code reproducing our analysis of all three models is available from the JAE data archive.

Acknowledgements

The research of Achim Zeileis was supported by the Austrian Science Foundation (FWF) under grant SFB#010. The work of Christian Kleiber was supported by the Deutsche Forschungsgemeinschaft (DFG), Sonderforschungsbereich 475.

We are grateful to Pierre Perron for detailed comments on a previous version which led us to provide a more complete explanation of the source of the problem in GAUSS. We would also like to thank Matei Demetrescu, Uwe Hassler, Roman Liesenfeld, and Mark Trede for running our GAUSS programs in GAUSS for Windows v5.0.22, v5.0.25, and v6.0.8.

REFERENCES

- Abramowitz M, Stegun IA. 1964. *Handbook of Mathematical Functions*. Washington, D.C.: U.S. Government Printing Office.
- Bai J. 1997. 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.
- Cody WJ. 1969. Rational Chebyshev approximations for the error function. *Mathematics of Computation* **22**: 631–637.
- Feller W. 1968. *An Introduction to Probability Theory and Its Applications, Vol. 1, 3rd ed.* New York: John Wiley.
- Leisch F, Rossini AJ. 2003. Reproducible statistical research. *Chance* **16**: 46–50.
- McCullough BD, Vinod HD. 2003. Verifying the solution from a nonlinear solver: A case study. *American Economic Review* **93**: 873–892.
- R Development Core Team. 2004. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-00-3.
URL <http://www.R-project.org/>
- Zeileis A, Kleiber C. 2004. Validating multiple structural change models – An extended case study. Report 12, Department of Statistics and Mathematics, Wirtschaftsuniversität Wien, Research Report Series.
URL <http://epub.wu-wien.ac.at/>
- 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**: 1–38.
URL <http://www.jstatsoft.org/v07/i02/>