Package: mbreaks (via r-universe)

August 23, 2024

Type Package

Title Estimation and inference for structural breaks in linear regression models

Version 1.0.0

Description The library contains efficient dynamic programming approach to conduct estimation, inference and testing for linear models in presence of structural breaks.

URL https://github.com/RoDivinity/mbreaks

BugReports https://github.com/RoDivinity/mbreaks/issues

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

Depends R (>= 3.3)

Imports ggplot2

Roxygen list(markdown = TRUE)

Suggests knitr, rmarkdown, graphics, testthat

VignetteBuilder knitr, rmarkdown

Repository https://rodivinity.r-universe.dev

RemoteUrl https://github.com/rodivinity/mbreaks

RemoteRef HEAD

RemoteSha 5e57109ba240611edb83e2d72504beab8d804040

Contents

compile_sbtests 2
correct 3
cvg 3
dating 4

diag_par . . . 4
dofix . . . 5
doglob . . . 6
doorder . . . 7
dorepart . . . 8
doseqtests . . . 9
dosequa . . . 11
dotest . . . 12
estim . . . 13
funcg . . . 14
getcv1 . . . 15
getcv2 . . . 15
interval . . . 16
jhatpr . . . 17
mdl . . . 17
mpower . . . 19
nkpc . . . 20
nldat . . . 20
onebp . . . 21
parti . . . 22
partione . . . 23
pftest . . . 23
plambda . . . 24
plot_model . . . 25
preparti . . . 25
print.model . . . 25
print.sbtests . . . 25
print.seqtests . . . 26
psigmq . . . 26
pvdel . . . 27
real . . . 27
sequa . . . 28
spflp1 . . . 28

Index **30**

`compile_sbtests` *Summary output of Sup Wald test*

Description

Function to format the output of the Sup F test

Usage

```
compile_sbtests(x, digits = -1, ...)
```

correct *Heteroskedasticy and autocorrelation consistency correction*

Description

function corrects the standard errors based on options of prewhitening using a AR(1) process estimation of error terms to obtain HAC errors/

Usage

```
correct(reg, res, prewhit)
```

Arguments

reg	matrix of regressors
res	residuals
prewhit	Option of using prewhitening process. If 1, an AR(1) process will be used to filter. If 0, skipped the filtering process

Value

hac Heteroskedasticy and autocorrelation consistent errors

cvg *Critical values computation*

Description

Function calculates the critical values for break date

Usage

```
cvg(eta, phi1s, phi2s)
```

Value

cvec Critical values of break dates

dating *Calculate optimal break points for pure structural change model*

Description

Function computes break points that globally minimizes SSR via dynamic programming approach. To avoid recursion depth increases as number of breaks in the model increases, a temporary array is used to store optimal partition with corresponding SSR for all permissible subsamples for all 1:m-1 breaks. For m breaks, the problem became finding where to insert the last feasible segments into a subsample to obtain minimum SSR for 1 to T

Usage

```
dating(y, z, h, m, q, bigT)
```

Arguments

y	dependent variable
z	matrix of independent variables
h	minimum length of segment
m	maximum number of breaks
q	number of regressors z
bigT	sample period T

Value

A list containing the following components:

glb	minimum global SSR
datevec	Vector of dates (optimal minimizers)
bigvec	Associated SSRs

diag_par *Diagonal partition given break dates*

Description

Function constructs the matrix of regressors z which coefficients are changed on the estimated break dates

Usage

```
diag_par(input, m, date)
```

Arguments

input	matrix of independent variables z with coefficients allowed to change overtime
m	number of breaks in the series
date	vector of estimated break dates

Value

output: matrix of partitioned variables corresponds to break dates

Examples

```
z = matrix(c(1:100),50,2)
m = 2  #2 breaks
date = matrix(c(15,30),2,1) #first break at t = 15; second break at t = 30
diag_par(z,m,date)
```

---

dofix *Estimate a model with pre-specified number of breaks*

Description

The following procedure constructs the so-called repartition estimates of the breaks obtained by the sequential method (see Bai (1995), Estimating Breaks one at a time, Econometric Theory, 13, 315-352. It alows estimates that have the same asymptotic distribution as those obtained by global minimization. Otherwise, the output from the procedure "estim" below do not deliver asymptotically correct confidence intervals for the break dates.

Usage

```
dofix(
  y_name,
  z_name = NULL,
  x_name = NULL,
  data,
  fixn = 5,
  eps = 1e-05,
  eps1 = 0.15,
  maxi = 10,
  fixb = 0,
  betaini = 0,
  printd = 0,
  prewhit = 1,
  robust = 1,
  hetdat = 1,
  hetvar = 1,
  hetq = 1,
```

```
  hetomega = 1,
  const = 1
)
```

Arguments

`eps`	convergence criterion for iterative recursive computation
`eps1`	trimming level
`maxi`	maximum number of iterations
`fixb`	option to use fixed initial input β. If `1`, the model will use values given in `betaini`. If `0`, betaini is skipped
`betaini`	Initial $beta_0$ to use in estimation
`printd`	option to print results of iterations for partial change model
`prewhit`	option to use AR(1) for prewhitening process
`robust`, `hetdat`, `hetvar`	options on error terms assumptions
`y`	dependent variables in matrix form
`z`	matrix of independent variables with coefficients are allowed to change across regimes
`x`	matrix of independent variables with coefficients constant across regimes
`m`	maximum number of breaks

Value

reparv Repartition method estimation of break dates

`doglob` *Global SSR minimizers procedure*

Description

A helper function to identify if the estimated break model is i) pure change or ii) partial change model. The procedure then calls appropriate functions to estimate the pure change model and to estimate the partial change model. This helper function is required for supF, UDMax, WDMax and supF(l+1|l) test functions invoked via

Usage

```
doglob(y, z, x, m, eps, eps1, maxi, fixb, betaini, printd)
```

Arguments

eps	convergence criterion for iterative recursive computation
eps1	trimming level
maxi	maximum number of iterations
fixb	option to use fixed initial input β. If 1, the model will use values given in betaini. If 0, betaini is skipped
betaini	Initial $beta_0$ to use in estimation
printd	option to print results of iterations for partial change model
y_name	dependent variables in matrix form
z_name	matrix of independent variables with coefficients are allowed to change across regimes
x_name	matrix of independent variables with coefficients constant across regimes

Value

A list containing the following components:

- glb Minimum global SSR
- datevec Vector of dates (optimal minimizers)
- bigvec Associated SSRs with possible break dates combination

---

doorder *Order estimation procedure*

Description

The function carry out the procedure to estimate order using BIC and the criterion of Liu, Wu and Zidek

Usage

```
doorder(
  y_name,
  z_name = NULL,
  x_name = NULL,
  data,
  m = 5,
  eps = 1e-05,
  eps1 = 0.15,
  maxi = 10,
  fixb = 0,
  betaini = 0,
  printd = 0,
  opt = "BIC",
  const = 1
)
```

Arguments

m	maximum number of breaks
eps	convergence criterion for iterative recursive computation
eps1	trimming level
maxi	maximum number of iterations
fixb	option to use fixed initial input β. If 1, the model will use values given in betaini. If 0, betaini is skipped
betaini	Initial $beta_0$ to use in estimation
printd	option to print results of iterations for partial change model
y	dependent variables in matrix form
z	matrix of independent variables with coefficients are allowed to change across regimes
x	matrix of independent variables with coefficients constant across regimes
bic	indicator which criterion is used in selecting number of breaks

Value

A list that contains following:

mBIC	number of breaks selected by BIC
mLWZ	number of breaks selected by LWZ

---

dorepart *Repartition procedure*

Description

The following procedure constructs the so-called repartition estimates of the breaks obtained by the sequential method (see Bai (1995), Estimating Breaks one at a time, Econometric Theory, 13, 315-352. It alows estimates that have the same asymptotic distribution as those obtained by global minimization. Otherwise, the output from the procedure "estim" below do not deliver asymptotically correct confidence intervals for the break dates.

Usage

```
dorepart(
  y_name,
  z_name = NULL,
  x_name = NULL,
  data,
  m = 5,
  eps = 1e-05,
  eps1 = 0.15,
  maxi = 10,
```

```
  fixb = 0,
  betaini = 0,
  printd = 0,
  prewhit = 1,
  robust = 1,
  hetdat = 1,
  hetvar = 1,
  const = 1,
  signif = 2
)
```

Arguments

m	maximum number of breaks
eps	convergence criterion for iterative recursive computation
eps1	trimming level
maxi	maximum number of iterations
fixb	option to use fixed initial input β. If 1, the model will use values given in betaini. If 0, betaini is skipped
betaini	Initial $beta_0$ to use in estimation
printd	option to print results of iterations for partial change model
prewhit	option to use AR(1) for prewhitening process
robust, hetdat, hetvar	options on error terms assumptions
y	dependent variables in matrix form
z	matrix of independent variables with coefficients are allowed to change across regimes
x	matrix of independent variables with coefficients constant across regimes

Value

reparv Repartition method estimation of break dates

`doseqtests` *SupF(l+1|l) test*

Description

Function computes the procedure of SupF(l+1|l) test. The function returns the test statistics of supF(l+1|l) test with null hypothesis is maximum number of break is l and alternative hypothesis is l+1. The l breaks under the null hypothesis are taken from the global minimization. Also, new date (if available) and critical values based on significant levels are returned for plotting and inference

Usage

```
doseqtests(
  y_name,
  z_name = NULL,
  x_name = NULL,
  data,
  m = 5,
  eps = 1e-05,
  eps1 = 0.15,
  maxi = 10,
  fixb = 0,
  betaini = 0,
  printd = 0,
  prewhit = 1,
  robust = 1,
  hetdat = 1,
  hetvar = 1,
  hetq = 1,
  hetomega = 1,
  const = 1
)
```

Arguments

`m`	maximum number of breaks
`eps`	convergence criterion for iterative recursive computation
`eps1`	trimming level
`maxi`	maximum number of iterations
`fixb`	option to use fixed initial input β. If `1`, the model will use values given in `betaini`. If `0`, betaini is skipped
`betaini`	Initial $beta_0$ to use in estimation
`printd`	option to print results of iterations for partial change model
`prewhit`	option to use AR(1) for prewhitening
`robust`, `hetdat`, `hetvar`	options on error terms assumptions
`y`	dependent variables in matrix form
`z`	matrix of independent variables with coefficients are allowed to change across regimes
`x`	matrix of independent variables with coefficients constant across regimes

Value

A list that contains following:

- supfl: SupF(l+1|l) test statistics

- cv: Critical values for SupF(l+1|l) test
- ndat: New date (if available)

`dosequa` *Sequential procedure*

Description

function to apply sequential procedure to obtain number of breaks and break dates. Current version only allows pure structural changes. This will be generalized

Usage

```
dosequa(
  y_name,
  z_name = NULL,
  x_name = NULL,
  data,
  m = 5,
  eps = 1e-05,
  eps1 = 0.15,
  maxi = 10,
  fixb = 0,
  betaini = 0,
  printd = 0,
  prewhit = 1,
  robust = 1,
  hetdat = 1,
  hetvar = 1,
  const = 1,
  signif = 2
)
```

Arguments

`m`	maximum number of breaks
`eps`	convergence criterion for iterative recursive computation
`eps1`	trimming level
`maxi`	maximum number of iterations
`fixb`	option to use fixed initial input β. If `1`, the model will use values given in `betaini`. If `0`, betaini is skipped
`betaini`	Initial $beta_0$ to use in estimation
`printd`	option to print results of iterations for partial change model
`prewhit`	option to use AR(1) for prewhitening process

`robust`, `hetdat`, `hetvar`
: options on error terms assumptions

`y`
: dependent variables in matrix form

`z`
: matrix of independent variables with coefficients are allowed to change across regimes

`x`
: matrix of independent variables with coefficients constant across regimes

Value

A list that contains following:

- nbreakNumber of breaks
- dateseqSequence of break dates

`dotest` *SupF, UDMax & WDMax testing procedure*

Description

The procedure calculate the test statistics and print results of the 2 main tests:

- SupF test F test of 0 vs m breaks
- Double Max test UDMax: the unweighted version and WDMax: the weighted version

Usage

```
dotest(
  y_name,
  z_name = NULL,
  x_name = NULL,
  data,
  m = 5,
  eps = 1e-05,
  eps1 = 0.15,
  maxi = 10,
  fixb = 0,
  betaini = 0,
  printd = 0,
  prewhit = 1,
  robust = 1,
  hetdat = 1,
  hetvar = 1,
  hetq = 1,
  hetomega = 1,
  const = 1
)
```

Arguments

y_name	name of dependent variable in the data set
z_name	name of independent variables in the data set which coefficients are allowed to change across regimes. default is vector of 1 (Mean-shift model)
x_name	name of independent variables in the data set which coefficients are constant across regimes. default is NULL
data	the data set for estimation
m	maximum number of breaks
eps	convergence criterion for iterative recursive computation
eps1	trimming level
maxi	maximum number of iterations
fixb	option to use fixed initial input β. If 1, the model will use values given in betaini. If 0, betaini is skipped
betaini	Initial $beta_0$ to use in estimation
printd	option to print results of iterations for partial change model
prewhit	option to use AR(1) for prewhitening
robust, hetdat, hetvar	options on error terms assumptions

Value

A list that contains following:

- ftest: SupF test of 0 vs m (1 to maximum) breaks statistics
- cv_supF: Critical values for Sup F test
- cv_Dmax: Critical values for Double Max test
- supF1: table summarizing the SupF test (for viewing purposes)
- UDMax: table summarizing the Double Max test (including UDMax statistics and CVs)

---

estim *Structural change model estimation*

Description

Function to estimate the model by OLS given the obtained break dates It also computes and reports confidence intervals for the break dates and corrected standard errors of coefficients estimates given specifications of errors covariance matrix via robust, hetomega, hetq, hetdat and hetvar

Usage

```
estim(m, q, z, y, b, robust, prewhit, hetomega, hetq, x, p, hetdat, hetvar)
```

Arguments

m	number of break
q	number of regressors z
z	matrix of regressor z with coefficients are allowed to change across regimes
b	break dates
prewhit	option to use prewhitening process based on AR(1) approximation
hetomega, hetq, hetdat, hetvar	options for assumptions on the error terms
x	matrix of regressor x with coefficients are constant across regimes
p	number of regressors x

Value

A list containing the following components:

- date List of estimated breaks
- CI List of Confidence Intervals for each corresponding break
- beta Estimated coefficients of the regression. The first (m+1)*q are coefficients of q variables z that change across regimes. The last p are coefficients of p variables x that are constant across regimes
- SE Corrected standard errors of the corresponding coefficients

---

funcg *Calculate p-value*

Description

Function computes the p-value of the test

Usage

```
funcg(x, bet, alph, b, deld, gam)
```

Value

g The p-value

getcv1 *SupF test Critical Values*

Description

Function to retrieve critical values of supF test stored in /SysData/SupF/cv_x.csv where x corresponds to the trimming level:

- 1: eps1 = 5%
- 2: eps1 = 10%
- 3: eps1 = 15%
- 4: eps1 = 20%
- 5: eps1 = 25%

The critical values are tabulated from @references

Usage

```
getcv1(signif, eps1)
```

Arguments

signif	significant level
eps1	trimming level

Value

cv Critical value of SupF test

getcv2 *SupF(l+1|l) test Critical Values*

Description

Function to retrieve critical values of SupF(l+1|l) test stored in /SysData/SupF_next/cv_x.csv where x corresponds to the trimming level:

- 1: eps1 = 5%
- 2: eps1 = 10%
- 3: eps1 = 15%
- 4: eps1 = 20%
- 5: eps1 = 25%

The critical values are tabulated from @references

Usage

```
getcv2(signif, eps1)
```

Arguments

signif	significant level
eps1	trimming level

Value

cv Critical value of SupF(l+1ll) test

---

interval *Break interval Function computes confidence intervals for the break dates*

Description

Function to compute confidence intervals for the break dates based on the "shrinking shifts" asymptotic framework

Usage

```
interval(y, z, zbar, b, q, m, robust, prewhit, hetomega, hetq, x, p)
```

Arguments

y	dependent variable
z	matrix of independent variables with coefficients allowed to change across regimes
zbar	partitioned matrix of independent variables with coefficients allowed to change across regimes according to break date
b	vector of estimated dates of breaks
q	number of regressors z
m	maximum number of breaks
robust, withb, hetdat, hetvar	options for assumptions of error terms structure
prewhit	Option of using prewhitening process. If 1, an AR(1) process will be used to filter. If 0, skipped the filtering process
x	matrix of independent variables with coefficients constant across regimes
p	number x regressors
bigT	sample period T

Value

bound Confidence intervals of break date in 90% and 95% significant level

jhatpr *Long-run covariance matrix computation*

Description

The procedure to compute the long run covariance matrix `jhat` based on variance matrix `vmat`

Usage

```
jhatpr(vmat)
```

Arguments

`vmat`	variance matrix

Value

jhat Long run covariance matrix

mdl *Robust structural change estimation*

Description

Function executes all the main procedures to estimate either i) pure structural breaks model or ii) partial structural breaks model

Usage

```
mdl(
  y_name,
  z_name = NULL,
  x_name = NULL,
  data,
  eps1 = 0.15,
  m = -1,
  prewhit = 1,
  robust = 1,
  hetdat = 1,
  hetvar = 1,
  hetomega = 1,
  hetq = 1,
  maxi = 10,
  eps = 1e-05,
  fixn = -1,
  printd = 0,
```

```
  const = 1,
  signif = 2
)
```

Arguments

y_name	name of dependent variable in the data set
z_name	name of independent variables in the data set which coefficients are allowed to change across regimes. default is vector of 1 (Mean-shift model)
x_name	name of independent variables in the data set which coefficients are constant across regimes. default is NULL
data	the data set for estimation
eps1	value of trimming (in percentage) for the construction and critical values. Minimal segment length will be set at default = int(eps1*T) (T is total sample size). • eps1=.05: Maximal value of m = 10 • eps1=.10: Maximal value of m = 8 • eps1=.15: Maximal value of m = 5 • eps1=.20: Maximal value of m = 3 • eps1=.25: Maximal value of m = 2 The default value is set at 0.15
m	Maximum number of structural changes allowed. If not specify, m will be set to default value from eps1
prewhit	set to 1 if want to apply AR(1) prewhitening prior to estimating the long run covariance matrix.
robust	set to 1 if want to allow for heterogeneity and autocorrelation in the residuals, 0 otherwise. The method used is *Andrews(1991)* automatic bandwidth with AR(1) approximation and the quadratic kernel. Note: Do not set to 1 if lagged dependent variables are included as regressors.
hetdat	option for the construction of the F tests. Set to 1 if want to allow different moment matrices of the regressors across segments. If hetdat = 0, the same moment matrices are assumed for each segment and estimated from the ful sample. It is recommended to set hetdat=1 if number of regressors x > 0.
hetvar	option for the construction of the F tests.Set to 1 if users want to allow for the variance of the residuals to be different across segments. If hetvar=0, the variance of the residuals is assumed constant across segments and constructed from the full sample. hetvar=1 when robust =1)
hetomega	used in the construction of the confidence intervals for the break dates. If hetomega=0, the long run covariance matrix of zu is assumed identical across segments (the variance of the errors u if robust=0)
hetq	used in the construction of the confidence intervals for the break dates. If hetq=0, the moment matrix of the data is assumed identical across segments
maxi	number of maximum iterations for recursive calculations of finding global minimizers.default = 10

eps	convergence criterion for recursive calculations
fixn	number of pre-specified breaks. default = -1. It will be replaced automatically to 2 if no specification is given
printd	Print option for model estimation. default = 0, to suppress intermediate outputs printing to console

Details

The 7 main procedures include:

- dotest: Procedure to conduct SupF test of 0 versus m breaks and Double Max test.
- doseqtests: Procedure to conduct sequential SupF(l+1|l) (l versus l+1 breaks)
- doorder: Procedure to find number of break by criteria selection (including BIC, LWZ and KT criterion)
- dosequa: Procedure to find number of break and break dates via sequential method
- dorepart: Procedure to find number of break and break dates via repartition method
- fix: Procedure to find number of break by pre-specified number of breaks, which are set at default to be 2

All default values of error assumptions (robust, hetdat, hetvar, hetq) will be set to 1. The implications on the structure of model's errors related to individual settings are explained within the arguments section for each option. Users can separately invoke only one at a time one of the main7 procedures mentioned above

Value

A list that contains the following:

- Wtest: Sup F tests of 0 versus m breaks and Double Max tests
- spflp1: Sequential Sup F test of l versus l+1 breaks
- BIC: Estimated number of breaks by BIC and the corresponding model
- LWZ: Estimated number of breaks by LWZ and the corresponding model
- KT: Estimated number of breaks by KT and the corresponding model
- sequa: Estimated number of breaks by sequential procedure the corresponding model
- repart: Estimated number of breaks by repartition procedure the corresponding model
- fix: Estimated model with pre-specified number of breaks

---

mpower *Function to calculate power of a matrix @noRd*

Description

Function to calculate power of a matrix @noRd

Usage

```
mpower(A, t)
```

nkpc *New Keynesian Phillips curve data*

Description

Data set from inflation and other macroeconomic variables

Usage

```
nkpc
```

Format

nkpc**:**

A data frame with 151 rows and 12 columns:

year Current period year

quarter Quarter in current period year

inf Inflation rate

inflag Inflation rate in previous period

inffut Expected inflation rate, taken as value of inflation rate of next period

ygap Productivity output gap

lbs

lbslag

spreadlag

dwlag

dcplag

Source

Perron, P. and Yamamoto, Y., 2015. "Using ols to estimate and test for structural changes in models with endogenous regressors." Journal of Applied Econometrics 30, 119–144.

nldat *Non-linear dating procedure*

Description

Function computes the break dates of a partial structural change model with pre-specified number of breaks m. The procedure is iterations of recursive computations of SSR and finding the m optimal global minimizers. The idea is similar to pure structural break case, with iteration to convergence to solve 2 sets of parameters.

Usage

```
nldat(y, z, x, h, m, p, q, bigT, fixb, eps, maxi, betaini, printd)
```

Arguments

y	time-series observations of dependent variable
z	matrix of regressors which coefficients are allowed to change across regimes
x	matrix of regressors which coefficients are constant across regime
p	number of z regressors
q	number of x regressors
bigT	the sample size T
fixb	option to use initial β If 1, procedure requires betaini. If 0, procedure will not use initial beta values
eps	Convergence criterion
betaini	initial beta values. Required when use with option fixb
printd	option to print output from iterated estimations. If 1, the results for each iteration will be printed in console log. If 0, no output will be printed

Value

A list containing the following components:

glb	minimum global SSR
datevec	Vector of dates (optimal minimizers)
bigvec	Associated SSRs

---

onebp *Optimal one break partition in partial structural change model*

Description

Function computes the optimal one break partition in partial structural change model by searching over all possible breaks given x regressors have unchanged coefficients. Iteration to convergence is used to deal with 2 sets of estimates needed to obtain: full-sample coefficients and regime-specific coefficients

Usage

```
onebp(y, z, x, h, start, last)
```

Arguments

y	dependent variables
z	independent variables with coefficients allowed to change across regimes
x	independent variables with constant coefficients across regimes
h	minimal length of segment
start	initial date to search
last	last date to search

Value

A list containing the following components

ssrmin	associated SSR of optimal break
dx	optimal date (global minimizer)

---

parti *Calculate optimal point for segment partition*

Description

Function calculates the optimal one break partition for a segment starting from start to last. The possible range of the break is within $[b_{last}, b_{end}]$ based on the vector of recursive SSR of the model

Usage

```
parti(start, b_start, b_end, last, bigvec, bigT)
```

Arguments

start	start date index of the segment
b_start	first possible break date
b_end	last possible breakdate
last	end date index of the segment
bigT	sample period T

Value

A list containing the following components:

ssrmin	associated SSR of optimal break
dx	optimal date (global minimizer)

partione *Optimal one break partitions with sequential method*

Description

Function calculates an optimal one break partitions for a segment that starts at date start and ends at date last. It returns the optimal break and the associated SSR. Procedure used with the sequential method when the T*(T+1)/2 vector of SSR is not computed.

Usage

```
partione(b1, b2, last, vssr, vssrev)
```

Arguments

last	end of segment considered
vssr	minimum SSRs of associated break date
vssrev	recursive SSRs of the model
b_start	first possible break date
b_end	last possible break date

Value

A list containing the following components:

ssrmin	associated SSR of optimal break
dx	optimal date (global minimizer)

pftest *SupF test for 0 vs i breaks*

Description

Function compute the supF test statistics of testing procedure with null hypothesis: no break versus alternative hypothesis: i breaks.

Function compute the supF test statistics of testing procedure with null hypothesis: no break versus alternative hypothesis: i breaks.

Usage

```
pftest(y, z, i, q, bigT, datevec, prewhit, robust, x, p, hetdat, hetvar)

pftest(y, z, i, q, bigT, datevec, prewhit, robust, x, p, hetdat, hetvar)
```

Arguments

y	dependent variables
z	independent variables with coefficients are allowed to change across regimes
i	number of breaks in the model
q	number of z regressors
bigT	sample period T
datevec	i estimated dates from the model
prewhit	Options for prewhitening process
robust, hetdat, hetvar	options for assumptions on error terms
x	independent variables with constant coefficients across regimes
p	number of x regressors

Value

ftest SupF test results

ftest SupF test results

---

plambda *Construct diagonal matrix according to break date*

Description

Function constructs a diagonal matrix of dimension (m+1) by (m+1) matrix with i-th entry $\frac{T_i - T_{i-1}}{T}$

Usage

```
plambda(b, m, bigT)
```

Arguments

b	Estimated date of changes
m	Number of breaks
bigT	The sample size T

Value

lambda (m+1) by (m+1) diagonal matrix with i-th entry $\frac{T_i - T_{i-1}}{T}$

plot_model *new plot function for class model*

Description

new plot function for class model

Usage

```
plot_model(model, CI = 0.95, title = NULL)
```

preparti *prepartion procedure*

Description

prepartion procedure

Usage

```
preparti(y, z, nbreak, dateseq, h, x, p)
```

print.model *Summary output of a n breaks model*

Description

Function to format the output of the n-break model

Usage

```
## S3 method for class 'model'
print(x, ...)
```

print.sbtests *S3 print function for sup F tests #'Function to format the output of the Sup F test*

Description

S3 print function for sup F tests #'Function to format the output of the Sup F test

Usage

```
## S3 method for class 'sbtests'
print(x, ...)
```

print.seqtests *S3 function to print sequential tests*

Description

S3 function to print sequential tests

Usage

```
## S3 method for class 'seqtests'
print(x, ...)
```

psigmq *Construct diagonal matrix of estimated variance*

Description

Function computes a diagonal matrix of dimension m+1 by m+1 with i-th entry is the estimated variance of residuals of segment i

Usage

```
psigmq(res, b, q, m, nt)
```

Arguments

res	big residual vector of the model
b	Estimated date of changes
m	Number of breaks
nt	The sample size

Value

sigmat (i+1)x(i+1) diagonal matrix with i-th entry equal to estimated variance of regime i

pvdel *Covariance matrix of estimator delta construction*

Description

Covariance matrix of estimator delta construction

Usage

```
pvdel(y, z, i, q, bigT, b, prewhit, robust, x, p, withb, hetdat, hetvar)
```

Arguments

y	dependent variable
z	matrix of independent variables with coefficients allowed to change across regimes
i	maximum number of breaks
q	number of regressors z
bigT	sample period T
b	vector of estimated dates of breaks
prewhit	Option of using prewhitening process. If 1, an AR(1) process will be used to filter. If 0, skipped the filtering process
robust, withb, hetdat, hetvar	options for assumptions of error terms structure
x	atrix of independent variables with constant coefficients across regimes
p	number x regressors

Value

vdel Covariance matrix of delta

real *World Health Organization TB data*

Description

Data set from the Garcia and Perron study's of ex-post exchange rate.

Usage

```
real
```

Format

`real`:

A data frame with 103 rows and 1 column:

rate Real exchange rate

Source

Garcia, R. and Perron, P., 1996. "An analysis of the real interest rate under regime shifts." Review of Economics and Statistics 111–125.

`sequa` *sequential procedure to obtain number of breaks and break dates*

Description

sequential procedure to obtain number of breaks and break dates

Usage

```
sequa(m, signif, q, h, bigT, robust, prewhit, z, y, x, p, hetdat, hetvar, eps1)
```

`spflp1` *SupF(l+1|l) test*

Description

Function computes the test statistics of supF(l+1|l) test with null hypothesis is l=nseg-1 and alternative hypothesis is l+1. The l breaks under the null hypothesis are taken from the global minimization.

Function computes the test statistics of supF(l+1|l) test with null hypothesis is l=nseg-1 and alternative hypothesis is l+1. The l breaks under the null hypothesis are taken from the global minimization.

Usage

```
spflp1(bigvec, dt, nseg, y, z, h, q, prewhit, robust, x, p, hetdat, hetvar)

spflp1(bigvec, dt, nseg, y, z, h, q, prewhit, robust, x, p, hetdat, hetvar)
```

Arguments

bigvec	associated SSR of estimated break date under H0
dt	estimated date under H0
nseg	number of segment under H1
z	independent variables with coefficients are allowed to change across regimes
q	number of z regressors
prewhit, robust, hetdat, hetvar	options on residuals/errors
x	independent variables with constant coefficients across regimes
p	number of x regressors

Value

A list that contains the following:

- maxfMaximum value of test
- newdAdditional date in alternative hypothesis

A list that contains the following:

- maxfMaximum value of test
- newdAdditional date in alternative hypothesis

Index

∗ **datasets**
    nkpc, 20
    real, 27

compile_sbtests, 2
correct, 3
cvg, 3

dating, 4
diag_par, 4
dofix, 5
doglob, 6
doorder, 7, *19*
dorepart, 8, *19*
doseqtests, 9, *19*
dosequa, 11, *19*
dotest, 12, *19*

estim, 13

funcg, 14

getcv1, 15
getcv2, 15

interval, 16

jhatpr, 17

mdl, 17
mpower, 19

nkpc, 20
nldat, 20

onebp, 21

parti, 22
partione, 23
pftest, 23
plambda, 24
plot_model, 25
preparti, 25
print.model, 25
print.sbtests, 25
print.seqtests, 26
psigmq, 26
pvdel, 27

real, 27

sequa, 28
spflp1, 28