Package ‘hystar’

July 22, 2025

Title Fit the Hysteretic Threshold Autoregressive Model

Version 1.0.0

Date 2023-07-03

Maintainer Daan de Jong <`daandejong94@gmail.com`>

Description Estimate parameters of the hysteretic threshold autoregressive (HysTAR) model, using conditional least squares.
In addition, you can generate time series data from the HysTAR model.
For details, see Li, Guan, Li and Yu (2015) <doi:10.1093/biomet/asv017>.

License MIT + file LICENSE

URL `https://github.com/daandejongen/hystar/`

BugReports `https://github.com/daandejongen/hystar/issues/`

Imports graphics, Rcpp, stats, utils

Suggests testthat (>= 3.0.0)

LinkingTo Rcpp

Encoding UTF-8

RoxygenNote 7.2.0

NeedsCompilation yes

Author Daan de Jong [aut, cre, cph] (ORCID: <`https://orcid.org/0000-0001-6034-0136`>),
European Research Council [fnd]

Repository CRAN

Date/Publication 2023-07-05 14:10:02 UTC

Contents

`hystar_fit` *Estimate the HysTAR model using conditional least squares estimation*

Description

This function allows you to estimate the parameters of the hysteretic threshold autoregressive (HysTAR) model.

Usage

```
hystar_fit(
  data,
  r = c(0.1, 0.9),
  d = 0L,
  p0 = 1L,
  p1 = 1L,
  p_select = "bic",
  thin = FALSE,
  tar = FALSE
)
```

Arguments

`data`	a vector, matrix or data.frame containing the outcome variable y in the first column and the threshold variable z in the second. Other columns are ignored. A vector, is taken to be both the outcome and control variable, so, in that case, a self-exciting HysTAR is fitted.
`r`	A vector or a matrix with search values for $\hat{r}_0, \hat{r}_1$. Defaults to `c(.1, .9)`. • A vector `r` must contain two values a and b in $[0, 1]$. The search space for the thresholds will be observed values of `z` between its $100a\%$ and $100b\%$ percentiles. • A matrix `r` allows for a custom search. It must have two columns, such that each row represents a pair $r_0 \leq r_1$ to test. You can use a matrix with one row if you don't want to estimate the thresholds. Note that the values in these matrix should be on the scale of `z`.
`d`	A numeric vector with one or more values for the search space of the delay parameter. Defaults to 1. Typically, d is not very large, so a reasonable search space might be 0, 1, 2, ..., 5.
`p0`	A numeric vector with one or more values for the search space of the autoregressive order of Regime 0. Defaults to 1.
`p1`	Same as `p0`, but for regime 1. Note that it does not need to be equal to `p0`.
`p_select`	The information criterion that should be minimized to select the orders p_0 and p_1. Choices: • `"aic"` (Akaike Information Criterion) • `"aicc"` (Corrected Akaike Information Criterion)

	• `"bic"` (default, Bayesian Information Criterion)
`thin`	`TRUE` (default) or `FALSE`. Only relevant when `r` is a vector. • If `TRUE` (default), the search space for the thresholds are the $100a\%, 100(a+0.01)\%, \ldots, 100b\%$ percentiles of `z`. This drastically reduces computation costs while keeping a reasonably large search space for the thresholds. Note that this is a purely practical choice with no theoretical justification. • If `FALSE`, all observed unique values of `z` between the $100a\%$ and $100b\%$ percentiles of `z` will be considered.
`tar`	`TRUE` or `FALSE` (default). Choose `TRUE` if you want to fit a traditional 2-regime threshold autoregressive (TAR) model. In this model, there is only one threshold (or equivalently, a HysTAR model with $r_0 = r_1$).

Details

In regime 0, y_t is predicted by values up to y_{t-p_0}. This implies that the first p_0 time points can not be predicted. E.g., if $p_0 = 2$, y_1 would miss a value from y_{-1}. Similarly, the value of the delay parameter implies that the regime is unknown for the first d time points. To ensure that the same data are used on all options for `d`, `p0` and `p1`, the first `max(d, p0, p1)` observations are discarded for estimation of the parameters.

Value

An object of S3 class `hystar_fit`, which is a `list` containing the following items:

- `$data`. A `data.frame` containing
  - `y`, the outcome variable
  - `z`, the threshold variable
  - `H`, a logical vector that indicates at which time points the hysteresis effect is happening. Note that this vector starts with `NA`(s), since not all values can be predicted in the HysTAR model. See Details.
  - `R`, the regime indicator vector. (Also starts with `NA`(s).)
- `$residuals`. Also accessible with the `residuals()` S3 method.
- `$coefficients`, a vector with the estimated coefficients. With the `coef()` S3 method, the coefficients are represented in a matrix. Use the `confint()` method to get the confidence intervals of the estimates.
- `$delay`, a scalar with the estimate for the delay parameter.
- `$thresholds`, a vector with the estimates of the thresholds.
- `$orders`, a vector with the estimates of the orders.
- `$resvar`, a vector with the estimates of the residual variances.
- `$rss`, the minimized residual sum of squares.
- `$ic`, a vector with the aic, the corrected aic and the bic.
- `$n`, a vector with the total effective observations and the effective obeservations in regime 0 and regime 1.
- `$eff`, a vector with the time indicators of the effective observations.
- `$equiv`, a matrix containing equivalent estimates for the delay and thresholds, i.e., estimates that imply exactly the same regime indicator vector, and as a result the same minimal residual sum of squares.

- `$r_search`, a vector with the r-values that were considered.
- `$tar`, Logical: `TRUE` if a TAR model was fitted.

Implemented generics for the `hystar_fit` class:

- `plot()` plots the `z` variable and the `y` variable above one another. Shading of the background visualizes the regimes. Thresholds are drawn as horizontal lines in the `z` plot. You can provide regime_names (char vector of 2), main (char vector of 1), xlab (char vector of 1) and ylab (char vector of 2).
- `summary()`, this also provides the p-values and standard errors for the estimates of the coefficients.
- `print()` prints the estimates within the mathematical representation of the model. Note that the scalar multiplied with `e[t]` is the standard deviation of the residuals, *not* the variance. See also the model definition above.
- `coef()`
- `confint()`
- `residuals()`
- `fitted()`
- `nobs()`

The HysTAR model

The HysTAR model is defined as:

$$y_t = \begin{cases} \phi_{00} + \phi_{01} y_{t-1} + \cdots + \phi_{0p_0} y_{t-p_0} + \sigma_0 \epsilon_t & \text{if } R_t = 0 \\ \phi_{10} + \phi_{11} y_{t-1} + \cdots + \phi_{1p_1} y_{t-p_1} + \sigma_1 \epsilon_t & \text{if } R_t = 1, \end{cases}$$

$$\text{with } R_t = \begin{cases} 0 & \text{if } z_{t-d} \in (-\infty, r_0] \\ R_{t-1} & \text{if } z_{t-d} \in (r_0, r_1] \\ 1 & \text{if } z_{t-d} \in (r_1, \infty), \end{cases}$$

where p_j denotes the order of regime $j \in \{0, 1\}$ with coefficients $\phi_{j0}, \ldots, \phi_{jp_j \in (-1,1)}$, σ_j is the standard deviation of the residuals, and $d \in \{0, 1, 2, \ldots\}$ is a delay parameter. The parameters of primary interest are the thresholds $r_0 \leq r_1$. We let $t = 0, 1, 2, ..., T$, where T is the number of observations.

Author(s)

Daan de Jong.

References

Li, Guodong, Bo Guan, Wai Keung Li, en Philip L. H. Yu. ‘Hysteretic Autoregressive Time Series Models’. Biometrika 102, nr. 3 (september 2015): 717–23.

Zhu, Ke, Philip L H Yu, en Wai Keung Li. ‘Testing for the Buffered Autoregressive Process’. Munich Personal RePEc Archive, (november 2013).

Examples

```
z <- z_sim(n_t = 200, n_switches = 5, start_regime = 1)
sim <- hystar_sim(z = z, r = c(-.5, .5), d = 2, phi_R0 = c(0, .6), phi_R1 = 1)
plot(sim)
fit <- hystar_fit(sim$data)
summary(fit)
```

`hystar_sim` *Simulate data from the HysTAR model*

Description

With this function, you can simulate observations from the HysTAR model, given its parameter values.

Usage

```
hystar_sim(z, r, d, phi_R0, phi_R1, resvar = c(1, 1), start_regime = NULL)
```

Arguments

`z`	A numeric vector representing the observed threshold variable. You can simulate `z` with `z_sim()`. Can not have missing values.
`r`	A numeric vector of length 2, representing the threshold values r_0 and r_1. The values must be inside the range of z, that is, larger than `min(z)` and smaller than `max(z)`. Otherwise, only one regime will be active so you might as well simulate an AR process, e.g. with `arima.sim()`. If you simulated `z` with `z_sim()` and `start_hyst` = `TRUE`, make sure to set the threshold values around the middle of the range of `z`, otherwise, the start will not be hysteretic.
`d`	A positive whole number representing the value of the delay parameter. It must be smaller than `length(z)`.
`phi_R0`	A vector containing the constant and autoregressive parameters $(\phi_0^{(0)}, \phi_1^{(0)}, \ldots, \phi_{p_0}^{(0)})$ of Regime 0. Note that the first value of this vector is *always* interpreted as the constant, so for an AR(1) process with no constant, you must use `phi_R0` = `c(0, .5)`, for example. Both orders must be smaller than `length(z)`. For valid standard errors of the estimates in `hystar_fit()`, the coefficients should imply that y is stationary, see Details.
`phi_R1`	The same as `phi_R0`, but for Regime 1.
`resvar`	A numeric vector of length 2 representing the variances of the residuals $\sigma^2_{(0)}$ and $\sigma^2_{(1)}$. The residuals are sampled from a normal distribution in the current implementation, but note that the model is defined for any i.i.d. vector of residuals with zero mean and finite variance.
`start_regime`	Optionally, a 0 or 1 that indicates which regime should be the first, in case the `z` variable starts in the hysteresis zone. This is only necessary when you use your 'own' `z` variable AND `z` starts in the hysteresis zone. A vector `z` simulated with `z_sim()` will contain information about if the start is hysteretic and what the starting regime is supposed to be (in the `attributes()` of `z`).

Details

Some details:

- To simulate `y`, 50 burn-in samples according the starting regime are used.
- The coefficients imply a stationary process of y_t if $\sum_{i=1}^{p_0} \phi_i^{(0)} < 1$ and $\sum_{i=1}^{p_1} \phi_i^{(1)} < 1$. See Zhu, Yu and Li (2013), p5.

Value

A list of class `hystar_sim` with elements

- `$data`, a `data.frame` with `length(z)` rows and 4 columns:
  - `y`, the outcome variable
  - `z`, the threshold variable
  - `H`, a logical vector that indicates at which time points the hysteresis effect is happening. Note that this vector starts with `NA`(s), since the first d time points have no values observed for z_{t-d}.
  - `R`, the regime indicator vector.
- `$r`, a numeric vector with the two threshold values,
- `$d`, the delay parameter,
- `$phi`, a numeric vector containing the coefficients. The names are such that `phi_R1_2` represents $\phi_2^{(1)}$, the second lag autoregressive coefficient in Regime 1,
- `$orders`, a numeric vector containing the two orders, and
- `$resvar`, a numeric vector with the residual variances of both regimes.

Implemented generics for the `hystar_sim` class:

- `plot()` plots the `z` variable and the `y` variable above one another. Shading of the background visualizes the regimes. Thresholds are drawn as horizontal lines in the `z` plot. You can provide regime_names (char vector of 2), main (char vector of 1), xlab (char vector of 1) and ylab (char vector of 2).
- `summary()` gives an overview of the true parameter values that were used.
- `print()` prints the parameter values within the mathematical representation of the model. Note that the scalar multiplied with `e[t]` is the standard deviation of the residuals, *not* the variance. See also the model definition above.

The HysTAR model

The HysTAR model is defined as:

$$y_t = \begin{cases} \phi_{00} + \phi_{01} y_{t-1} + \cdots + \phi_{0p_0} y_{t-p_0} + \sigma_0 \epsilon_t & \text{if } R_t = 0 \\ \phi_{10} + \phi_{11} y_{t-1} + \cdots + \phi_{1p_1} y_{t-p_1} + \sigma_1 \epsilon_t & \text{if } R_t = 1, \end{cases}$$

$$\text{with } R_t = \begin{cases} 0 & \text{if } z_{t-d} \in (-\infty, r_0] \\ R_{t-1} & \text{if } z_{t-d} \in (r_0, r_1] \\ 1 & \text{if } z_{t-d} \in (r_1, \infty), \end{cases}$$

where p_j denotes the order of regime $j \in \{0, 1\}$ with coefficients $\phi_{j0}, \ldots, \phi_{jp_j \in (-1,1)}$, σ_j is the standard deviation of the residuals, and $d \in \{0, 1, 2, \ldots\}$ is a delay parameter. The parameters of primary interest are the thresholds $r_0 \leq r_1$. We let $t = 0, 1, 2, ..., T$, where T is the number of observations.

Author(s)

Daan de Jong.

References

Li, Guodong, Bo Guan, Wai Keung Li, en Philip L. H. Yu. 'Hysteretic Autoregressive Time Series Models'. Biometrika 102, nr. 3 (september 2015): 717–23.

Zhu, Ke, Philip L H Yu, en Wai Keung Li. 'Testing for the Buffered Autoregressive Process'. Munich Personal RePEc Archive, (november 2013).

Examples

```
z <- z_sim(n_t = 200, n_switches = 5, start_regime = 1)
sim <- hystar_sim(z = z, r = c(-.5, .5), d = 2, phi_R0 = c(0, .6), phi_R1 = 1)
plot(sim)
fit <- hystar_fit(sim$data)
summary(fit)
```

`z_sim` *Simulate the threshold/control variable Z*

Description

This is a function you can use to simulate time series data for a threshold variable of the HysTAR model. The time series is a (co)sine wave, such that thresholds are crossed in a predictable way. This function is designed to be used in combination with `hystar_sim()`.

Usage

```
z_sim(n_t, n_switches, start_regime = 0, start_hyst = FALSE, range = c(-1, 1))
```

Arguments

`n_t`	The desired length of the simulated time series of `z`. The actual vector that is returned will contain 10 more time points, see Details. Note that `n_t` will also be the length of `y`, when you feed `z` to `hystar_sim`.
`n_switches`	A scalar indicating the desired number of regime switches. Basically, it is the number of times the variable moves to (and reaches) its minimum or to its maximum. If the thresholds are within the range of `z`, as they should, this will guarantee the same number of regime switches when the delay parameter of the HysTAR model is greater or equal than the highest order. See Details.
`start_regime`	The starting regime of the HysTAR model, 0 (default) or 1.

`start_hyst`	Logical, should `z` start in the hysteresis zone? Of course, this also depends on `r`, and `r` is not yet specified in this function. Rather, setting `start_hyst` to `TRUE` makes `z` start at in the middle of its range, which makes it easy to set the threshold values "around" the first values of `z`.
`range`	A numeric vector of length 2 indicating the desired range (min, max) of `z`.

Details

The first value of `y` that can be predicted in the HysTAR model is at time point $\max\{d, p\} + 1$, where $p = \max\{p_0, p_1\}$. This is because we need to observe y_{t-p} and z_{t-d}. So the first observed value of `z` that determines a regime is at time point $\max\{d, p\} + 1 - d$. To make sure that this time point corresponds to the start that you request, `z_sim()` starts with 10 extra time points. In this way, `hystar_sim` can select the appropriate time points, based on `d` and `p0`, `p1`.

Value

A numeric vector of length `n_t`. This vector has two attributes `"start_regime"` and `"start_hyst"` corresponding to the values you provided. These attributes are used by `hystar_sim()`.

The HysTAR model

The HysTAR model is defined as:

$$y_t = \begin{cases} \phi_{00} + \phi_{01} y_{t-1} + \cdots + \phi_{0p_0} y_{t-p_0} + \sigma_0 \epsilon_t & \text{if } R_t = 0 \\ \phi_{10} + \phi_{11} y_{t-1} + \cdots + \phi_{1p_1} y_{t-p_1} + \sigma_1 \epsilon_t & \text{if } R_t = 1, \end{cases}$$

$$\text{with } R_t = \begin{cases} 0 & \text{if } z_{t-d} \in (-\infty, r_0] \\ R_{t-1} & \text{if } z_{t-d} \in (r_0, r_1] \\ 1 & \text{if } z_{t-d} \in (r_1, \infty), \end{cases}$$

where p_j denotes the order of regime $j \in \{0, 1\}$ with coefficients $\phi_{j0}, \dots, \phi_{jp_j \in (-1,1)}$, σ_j is the standard deviation of the residuals, and $d \in \{0, 1, 2, \dots\}$ is a delay parameter. The parameters of primary interest are the thresholds $r_0 \leq r_1$. We let $t = 0, 1, 2, ..., T$, where T is the number of observations.

Examples

```
z <- z_sim(n_t = 200, n_switches = 5, start_regime = 1)
sim <- hystar_sim(z = z, r = c(-.5, .5), d = 2, phi_R0 = c(0, .6), phi_R1 = 1)
plot(sim)
fit <- hystar_fit(sim$data)
summary(fit)
```

Index