

# Package ‘toeclim’

October 10, 2026

**Type** Package

**Title** Time of Emergence of Climate Signals from Internal Variability

**Version** 0.1.0

**Description** Detects when a forced climate change signal emerges from internal variability ('time of emergence') in annual climate time series such as temperature or precipitation. The signal is estimated with a linear, quadratic or locally weighted smoother and compared with noise estimated from a baseline period. Emergence is defined through configurable signal-to-noise thresholds and persistence rules. Uncertainty in the time of emergence is quantified by autoregressive or moving-block bootstrap resampling. Helpers are provided for multi-member ensembles, threshold sensitivity, anomaly and annual aggregation, Mann-Kendall trend tests with trend-free prewhitening, and synthetic data generation. The approach builds on Hawkins and Sutton (2012) <[doi:10.1029/2011GL050087](https://doi.org/10.1029/2011GL050087)>, Yue et al. (2002) <[doi:10.1002/hyp.1095](https://doi.org/10.1002/hyp.1095)> and Sen (1968) <[doi:10.1080/01621459.1968.10480934](https://doi.org/10.1080/01621459.1968.10480934)>.

**License** MIT + file LICENSE

**Depends** R (>= 3.5.0)

**Imports** graphics, grDevices, stats

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0)

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.3.1

**NeedsCompilation** no

**Author** Abul Kashem Faruki Fahim [aut, cre, cph] (ORCID: <<https://orcid.org/0000-0002-8554-4730>>)

**Maintainer** Abul Kashem Faruki Fahim <[farukifahim@du.ac.bd](mailto:farukifahim@du.ac.bd)>

**Repository** CRAN

**Date/Publication** 2026-10-10 10:50:02 UTC

Contents

|                             |           |
|-----------------------------|-----------|
| plot.toe . . . . .          | 2         |
| plot.toe_ensemble . . . . . | 3         |
| toe_annual . . . . .        | 4         |
| toe_anomaly . . . . .       | 5         |
| toe_emerge . . . . .        | 6         |
| toe_ensemble . . . . .      | 8         |
| toe_example . . . . .       | 9         |
| toe_noise . . . . .         | 10        |
| toe_prob . . . . .          | 11        |
| toe_sensitivity . . . . .   | 12        |
| toe_simulate . . . . .      | 13        |
| toe_trend . . . . .         | 14        |
| <b>Index</b>                | <b>15</b> |

---

|          |                                          |
|----------|------------------------------------------|
| plot.toe | <i>Plot a time-of-emergence analysis</i> |
|----------|------------------------------------------|

---

Description

Draws the observed series, the fitted signal, the noise envelope (baseline mean +/- k \* sigma) and the time of emergence. If a bootstrap was run, its interval is shaded.

Usage

```
## S3 method for class 'toe'
plot(x, xlab = "Time", ylab = "Value", main = "Time of emergence", ...)
```

Arguments

- x                    A toe object from [toe\\_emerge\(\)](#).
- xlab, ylab          Axis labels.
- main                Title.
- ...                 Further arguments passed to [graphics::plot\(\)](#).

Value

x, invisibly.

Examples

```
d <- toe_simulate(n_years = 150, slope = 0.03, onset = 1970, seed = 8)
plot(toe_emerge(d$member_01, d$year, baseline = c(1900, 1950)))
```

---

|                   |                                        |
|-------------------|----------------------------------------|
| plot.toe_ensemble | <i>Plot ensemble time of emergence</i> |
|-------------------|----------------------------------------|

---

## Description

Shows every member (thin lines), the ensemble mean, the pooled emergence time and a rug of the individual member emergence times.

## Usage

```
## S3 method for class 'toe_ensemble'
plot(
  x,
  xlab = "Time",
  ylab = "Value",
  main = "Ensemble time of emergence",
  ...
)
```

## Arguments

|            |                                                             |
|------------|-------------------------------------------------------------|
| x          | A toe_ensemble object from <code>toe_ensemble()</code> .    |
| xlab, ylab | Axis labels.                                                |
| main       | Title.                                                      |
| ...        | Further arguments passed to <code>graphics::plot()</code> . |

## Value

x, invisibly.

## Examples

```
d <- toe_simulate(n_years = 150, onset = 1970, n_members = 6, seed = 9)
ens <- toe_ensemble(d[, grep("^member", names(d))], d$year,
  baseline = c(1900, 1950))
plot(ens)
```

---

|            |                                                   |
|------------|---------------------------------------------------|
| toe_annual | <i>Aggregate sub-annual data to annual values</i> |
|------------|---------------------------------------------------|

---

## Description

Collapses monthly, daily or other sub-annual observations to one value per calendar year. Years with too few valid observations are set to NA so that partial years do not bias the annual statistic.

## Usage

```
toe_annual(x, dates, fun = mean, min_frac = 0.8)
```

## Arguments

|          |                                                                                                                                             |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------|
| x        | Numeric vector of observations.                                                                                                             |
| dates    | Vector of the same length as x, of class Date or POSIXt.                                                                                    |
| fun      | Function used to aggregate within each year. Defaults to <code>base::mean()</code> . Use <code>base::sum()</code> for precipitation totals. |
| min_frac | Minimum fraction (0 to 1) of the maximum number of observations per year that must be non-missing for a year to be kept. Defaults to 0.8.   |

## Value

A data frame with columns year, value (the aggregated value, NA for incomplete years) and n (number of valid observations).

## See Also

[toe\\_anomaly\(\)](#)

## Examples

```
dates <- seq(as.Date("2000-01-01"), as.Date("2004-12-01"), by = "month")
x <- rnorm(length(dates), mean = 10)
x[c(3, 4, 5, 6, 7, 8)] <- NA # a gappy year
toe_annual(x, dates)
```

---

toe\_anomaly*Compute anomalies relative to a baseline climatology*

---

**Description**

Subtracts a climatological mean, computed over a baseline period, from every observation. The climatology can be a single mean or can be resolved by calendar month or day of year to remove the seasonal cycle.

**Usage**

```
toe_anomaly(x, time, baseline, by = c("none", "month", "day"))
```

**Arguments**

|          |                                                                                                            |
|----------|------------------------------------------------------------------------------------------------------------|
| x        | Numeric vector of observations.                                                                            |
| time     | Either numeric years (only with by = "none") or a vector of class Date or POSIXt.                          |
| baseline | Numeric vector c(start_year, end_year) (inclusive) defining the reference period.                          |
| by       | Resolution of the climatology: "none" (one mean), "month" (12 monthly means) or "day" (day-of-year means). |

**Value**

A data frame with columns time, value, climatology and anomaly.

**See Also**

[toe\\_annual\(\)](#)

**Examples**

```
dates <- seq(as.Date("1980-01-01"), as.Date("2019-12-01"), by = "month")
x <- 10 + 8 * sin(2 * pi * (as.integer(format(dates, "%m")) - 1) / 12) +
  rnorm(length(dates))
a <- toe_anomaly(x, dates, baseline = c(1981, 2010), by = "month")
head(a)
```

toe\_emerge

*Time of emergence of a climate signal***Description**

Estimates when a forced signal becomes distinguishable from internal variability in a single time series. The series is decomposed into a smooth *signal* (relative to the baseline mean) and *noise* (standard deviation estimated from the baseline period or from residuals). The **time of emergence** is the first time after the baseline at which the signal-to-noise ratio  $|S/N|$  reaches  $k$  and persistence criteria are met.

**Usage**

```
toe_emerge(
  x,
  time = NULL,
  baseline,
  signal = c("quadratic", "linear", "loess"),
  span = 0.5,
  k = 2,
  persist = 5L,
  sustain = c("window", "end"),
  noise = c("baseline", "residual"),
  direction = c("both", "positive", "negative"),
  n_boot = 0L,
  boot = c("ar1", "block"),
  block_length = NULL,
  conf_level = 0.9,
  seed = NULL
)
```

**Arguments**

|          |                                                                                                                                                    |
|----------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| x        | Numeric vector, typically annual anomalies or annual means. Missing values are dropped with a warning.                                             |
| time     | Numeric vector of times (e.g. years), strictly increasing. If NULL, seq_along(x) is used.                                                          |
| baseline | Numeric vector c(start, end) (in the units of time) defining the reference period used for the reference mean and, by default, the noise estimate. |
| signal   | Signal estimator: "quadratic" (default), "linear" or "loess". See the section <i>Choosing the signal model</i> .                                   |
| span     | Smoothing span for signal = "loess". Default 0.5.                                                                                                  |
| k        | Signal-to-noise threshold in units of noise standard deviations. Default 2. Hawkins and Sutton (2012) use $k = 2$ ; IPCC AR5 used $k = 1$ .        |
| persist  | Minimum number of consecutive time steps for which the threshold must be exceeded. Default 5.                                                      |

|              |                                                                                                                                                                                                                              |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| sustain      | How persistence is judged. "window" (default) accepts the first run of at least persist exceedances. "end" additionally requires the exceedance to hold until the end of the record (a conservative, "permanent" emergence). |
| noise        | Noise estimator. "baseline" (default) uses the detrended standard deviation within the baseline period; "residual" uses residuals of the signal fit over the whole record.                                                   |
| direction    | Direction of change that counts as emergence: "both" (default), "positive" or "negative".                                                                                                                                    |
| n_boot       | Number of bootstrap replicates used to quantify uncertainty in the time of emergence. 0 (default) skips the bootstrap.                                                                                                       |
| boot         | Bootstrap scheme: "ar1" (parametric AR(1) residuals; default) or "block" (moving-block resampling of residuals).                                                                                                             |
| block_length | Block length for boot = "block". Defaults to $\text{ceiling}(n^{(1/3)})$ .                                                                                                                                                   |
| conf_level   | Confidence level for the bootstrap interval. Default 0.9.                                                                                                                                                                    |
| seed         | Optional integer seed. The global random number state is restored on exit.                                                                                                                                                   |

### Details

The bootstrap resamples residuals around the fitted signal, adds them back to the signal, and repeats the entire procedure (signal fit, noise estimate and emergence rule). The resulting distribution of emergence times includes NA for replicates that never emerge; the proportion that emerge is reported as prob\_emerged. The interval is computed from the replicates that do emerge.

Persistence is counted in observations, so time should be regularly spaced (e.g. annual).

### Value

An object of class toe, a list with elements

toe Time of emergence, or NA if the signal never emerges.

data Data frame with time, value, signal (fit minus baseline mean), snr and exceed.

sigma Noise standard deviation used.

baseline\_mean Baseline mean of x.

boot NULL, or a list with toe (vector of replicate times), ci, prob\_emerged, conf\_level and method.

params The settings used.

### Choosing the signal model

Forced climate responses are rarely straight lines: they are often flat and then accelerate. A straight-line fit to such a record is too high early on and can cross the noise threshold *before* the change actually begins. In a simulation study bundled with the package's development (200 series, true onset 1970, true emergence about 1990), a linear signal produced a median emergence year of 1971 and emerged before onset in 35% of series, whereas quadratic and loess signals gave medians of 1990 and 1989 and never emerged before onset. "quadratic" is therefore the default. Use "linear" only when a constant rate of change is plausible.

## References

Hawkins, E. and Sutton, R. (2012) [doi:10.1029/2011GL050087](https://doi.org/10.1029/2011GL050087).

## See Also

[toe\\_ensemble\(\)](#), [toe\\_sensitivity\(\)](#), [toe\\_prob\(\)](#), [toe\\_noise\(\)](#), [toe\\_simulate\(\)](#)

## Examples

```
d <- toe_simulate(n_years = 150, slope = 0.03, onset = 1970,
                 sigma = 0.3, phi = 0.5, seed = 10)
fit <- toe_emerge(d$member_01, d$year, baseline = c(1900, 1950))
fit

# With bootstrap uncertainty (small n_boot to keep the example fast)
fit_b <- toe_emerge(d$member_01, d$year, baseline = c(1900, 1950),
                   n_boot = 100, seed = 1)

fit_b$boot$ci
plot(fit_b)
```

---

|              |                                             |
|--------------|---------------------------------------------|
| toe_ensemble | <i>Time of emergence across an ensemble</i> |
|--------------|---------------------------------------------|

---

## Description

Applies [toe\\_emerge\(\)](#) to every member of a multi-member ensemble (e.g. a climate model initial-condition large ensemble) and summarises the spread of emergence times. Optionally also estimates a *pooled* emergence time from the ensemble-mean signal against the average single-member noise.

## Usage

```
toe_ensemble(x, time = NULL, baseline, pooled = TRUE, ...)
```

## Arguments

|          |                                                                                                                                                                 |
|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| x        | A numeric matrix or data frame with one row per time step and one column per ensemble member.                                                                   |
| time     | Numeric vector of times with length <code>nrow(x)</code> . If <code>NULL</code> , <code>seq_len(nrow(x))</code> is used.                                        |
| baseline | Numeric vector <code>c(start, end)</code> ; see <a href="#">toe_emerge()</a> .                                                                                  |
| pooled   | Logical. Also estimate the pooled ensemble-mean emergence? Default <code>TRUE</code> .                                                                          |
| ...      | Further arguments passed to <a href="#">toe_emerge()</a> , e.g. <code>k</code> , <code>persist</code> , <code>signal</code> . Do not pass <code>n_boot</code> . |

**Value**

An object of class `toe_ensemble`, a list with

- `members` Data frame with member, toe and sigma.
- `summary` Named vector: n, n\_emerged, median, min, max, q05, q95 of member emergence times (over members that emerge).
- `pooled` A toe object for the ensemble mean (or NULL).
- `data` Data frame of time and the input members.

**See Also**

[toe\\_emerge\(\)](#)

**Examples**

```
d <- toe_simulate(n_years = 150, slope = 0.03, onset = 1970,
                 n_members = 8, seed = 4)
ens <- toe_ensemble(d[, grep("^member", names(d))], d$year,
                  baseline = c(1900, 1950))
ens
```

---

toe\_example

---

*Synthetic 10-member ensemble of annual temperature anomalies*


---

**Description**

A synthetic data set that mimics a regional annual-mean temperature anomaly from a single-model initial-condition large ensemble. It is generated with [toe\\_simulate\(\)](#) and is intended for examples, tests and teaching; it does not represent any real model or observation.

**Usage**

```
toe_example
```

**Format**

A data frame with 201 rows (1900-2100) and 12 variables:

**year** Calendar year.

**forced** The true forced response (degrees C), known by construction.

**member\_01, member\_02, member\_03, member\_04, member\_05, member\_06, member\_07, member\_08, member\_09**  
Annual anomaly (degrees C) for each ensemble member.

**Details**

The forced response is flat until 1980 and then warms at 0.03 degrees C per year. Noise is AR(1) with standard deviation 0.35 and lag-1 autocorrelation 0.6.

**Source**

Simulated; see data-raw/toe\_example.R.

**See Also**

[toe\\_simulate\(\)](#), [toe\\_ensemble\(\)](#)

**Examples**

```
head(toe_example[, 1:4])
```

---

|           |                                              |
|-----------|----------------------------------------------|
| toe_noise | <i>Estimate internal variability (noise)</i> |
|-----------|----------------------------------------------|

---

**Description**

Removes a smooth signal from a series and characterises what is left: its standard deviation, lag-1 autocorrelation and the effective number of independent samples.

**Usage**

```
toe_noise(
  x,
  time = NULL,
  signal = c("quadratic", "linear", "loess"),
  span = 0.5
)
```

**Arguments**

|        |                                                                                                     |
|--------|-----------------------------------------------------------------------------------------------------|
| x      | Numeric vector.                                                                                     |
| time   | Numeric vector of times, strictly increasing. If NULL, seq_along(x) is used.                        |
| signal | Signal model removed before computing noise statistics: "quadratic" (default), "linear" or "loess". |
| span   | Smoothing span for signal = "loess". Default 0.5.                                                   |

**Value**

A list with elements sigma (residual standard deviation, adjusted for the degrees of freedom used by the signal model), phi (lag-1 autocorrelation), n\_eff (effective sample size,  $n * (1 - \phi) / (1 + \phi)$ , capped at n), residuals and signal.

**See Also**

[toe\\_trend\(\)](#), [toe\\_emerge\(\)](#)

**Examples**

```
d <- toe_simulate(n_years = 100, sigma = 0.3, phi = 0.6, seed = 42)
str(toe_noise(d$member_01, d$year)[c("sigma", "phi", "n_eff")])
```

---

|          |                                                 |
|----------|-------------------------------------------------|
| toe_prob | <i>Probability of emergence by a given time</i> |
|----------|-------------------------------------------------|

---

**Description**

Uses the bootstrap distribution stored in a `toe_emerge()` result to estimate the probability that the signal has emerged by each of the supplied times,  $P(\text{ToE} \leq t)$ . Replicates that never emerge count as "not emerged".

**Usage**

```
toe_prob(object, times)
```

**Arguments**

|        |                                                               |
|--------|---------------------------------------------------------------|
| object | A toe object created with <code>n_boot &gt; 0</code> .        |
| times  | Numeric vector of times at which to evaluate the probability. |

**Value**

A data frame with columns `time` and `prob`.

**See Also**

`toe_emerge()`

**Examples**

```
d <- toe_simulate(n_years = 150, slope = 0.03, onset = 1970, seed = 3)
f <- toe_emerge(d$member_01, d$year, baseline = c(1900, 1950),
               n_boot = 100, seed = 2)
toe_prob(f, times = c(1990, 2000, 2010, 2030))
```

---

|                 |                                                                  |
|-----------------|------------------------------------------------------------------|
| toe_sensitivity | <i>Sensitivity of the time of emergence to threshold choices</i> |
|-----------------|------------------------------------------------------------------|

---

### Description

Re-estimates emergence over a grid of signal-to-noise thresholds  $k$  and persistence lengths, making the dependence on these subjective choices explicit.

### Usage

```
toe_sensitivity(
  x,
  time = NULL,
  baseline,
  k = c(1, 2, 3),
  persist = c(1, 5, 10),
  ...
)
```

### Arguments

|                       |                                                                                                                                                                     |
|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>x</code>        | Numeric vector, typically annual anomalies or annual means. Missing values are dropped with a warning.                                                              |
| <code>time</code>     | Numeric vector of times (e.g. years), strictly increasing. If <code>NULL</code> , <code>seq_along(x)</code> is used.                                                |
| <code>baseline</code> | Numeric vector <code>c(start, end)</code> (in the units of time) defining the reference period used for the reference mean and, by default, the noise estimate.     |
| <code>k</code>        | Numeric vector of thresholds.                                                                                                                                       |
| <code>persist</code>  | Numeric vector of persistence lengths.                                                                                                                              |
| <code>...</code>      | Further arguments passed to <a href="#">toe_emerge()</a> (e.g. <code>signal</code> , <code>noise</code> , <code>sustain</code> ). Do not pass <code>n_boot</code> . |

### Value

A data frame with columns `k`, `persist` and `toe`.

### See Also

[toe\\_emerge\(\)](#)

### Examples

```
d <- toe_simulate(n_years = 150, slope = 0.03, onset = 1970, seed = 5)
toe_sensitivity(d$member_01, d$year, baseline = c(1900, 1950),
  k = c(1, 2, 3), persist = c(1, 5, 10))
```

---

|              |                                                                     |
|--------------|---------------------------------------------------------------------|
| toe_simulate | <i>Simulate synthetic climate series with a known forced signal</i> |
|--------------|---------------------------------------------------------------------|

---

### Description

Generates annual series consisting of a piecewise-linear forced response (flat until onset, then rising at slope per year) plus stationary AR(1) noise. Useful for testing emergence methods against a known truth.

### Usage

```
toe_simulate(
  n_years = 150,
  start = 1900,
  slope = 0.03,
  onset = start + 60,
  sigma = 0.3,
  phi = 0.5,
  n_members = 1,
  seed = NULL
)
```

### Arguments

|           |                                                                                |
|-----------|--------------------------------------------------------------------------------|
| n_years   | Number of years. Default 150.                                                  |
| start     | First year. Default 1900.                                                      |
| slope     | Forced trend after onset, in units per year. Default 0.03.                     |
| onset     | Year at which the trend begins. Default start + 60.                            |
| sigma     | Marginal standard deviation of the noise. Default 0.3.                         |
| phi       | Lag-1 autocorrelation of the noise, between -1 and 1 (exclusive). Default 0.5. |
| n_members | Number of independent noise realisations (ensemble members). Default 1.        |
| seed      | Optional integer seed; the global random state is restored.                    |

### Value

A data frame with columns year, forced (the true forced response) and one column per member named member\_01, member\_02, ...

### See Also

[toe\\_example](#), [toe\\_emerge\(\)](#)

### Examples

```
d <- toe_simulate(n_years = 100, n_members = 3, seed = 1)
head(d)
```

toe\_trend

*Robust trend test: Sen's slope with a Mann-Kendall test***Description**

Estimates the linear trend with the Theil-Sen estimator and tests its significance with the Mann-Kendall test. Because serial correlation inflates false-positive rates, the test optionally applies *trend-free prewhitening* (TFPW; Yue et al., 2002) when the lag-1 autocorrelation of the detrended series is significant.

**Usage**

```
toe_trend(x, time = NULL, conf_level = 0.95, prewhiten = TRUE)
```

**Arguments**

|            |                                                                                                 |
|------------|-------------------------------------------------------------------------------------------------|
| x          | Numeric vector.                                                                                 |
| time       | Numeric vector of times (e.g. years), strictly increasing. If NULL, seq_along(x) is used.       |
| conf_level | Confidence level for the slope interval. Default 0.95.                                          |
| prewhiten  | Logical; apply trend-free prewhitening when lag-1 autocorrelation is significant? Default TRUE. |

**Value**

An object of class `toe_trend`, a list with elements `slope`, `intercept`, `slope_ci`, `conf_level`, `S` (Mann-Kendall statistic), `z`, `p_value`, `n`, `r1` (lag-1 autocorrelation of detrended data), `prewhitened` (logical) and `call`. The slope is in units of x per unit of time.

**References**

Sen (1968) [doi:10.1080/01621459.1968.10480934](https://doi.org/10.1080/01621459.1968.10480934); Yue et al. (2002) [doi:10.1002/hyp.1095](https://doi.org/10.1002/hyp.1095).

**See Also**

[toe\\_noise\(\)](#), [toe\\_emerge\(\)](#)

**Examples**

```
d <- toe_simulate(n_years = 80, slope = 0.02, onset = 1900, seed = 1)
toe_trend(d$member_01, d$year)
```

# Index

## \* datasets

toe\_example, 9

base::mean(), 4

base::sum(), 4

graphics::plot(), 2, 3

plot.toe, 2

plot.toe\_ensemble, 3

toe\_annual, 4

toe\_annual(), 5

toe\_anomaly, 5

toe\_anomaly(), 4

toe\_emerge, 6

toe\_emerge(), 2, 8–14

toe\_ensemble, 8

toe\_ensemble(), 3, 8, 10

toe\_example, 9, 13

toe\_noise, 10

toe\_noise(), 8, 14

toe\_prob, 11

toe\_prob(), 8

toe\_sensitivity, 12

toe\_sensitivity(), 8

toe\_simulate, 13

toe\_simulate(), 8–10

toe\_trend, 14

toe\_trend(), 10