Package: edetect (via r-universe)

October 22, 2026

Title Anytime-Valid Control Charts with E-Detectors

Version 0.1.0.9000

Description Sequential change detection for process monitoring with the e-detectors of Shin, Ramdas and Rinaldo (2024) <doi:10.51387/23-NEJSDS51>: nonparametric, anytime-valid analogues of the CUSUM and Shiryaev-Roberts charts whose average run length to a false alarm is guaranteed for every distribution in a declared pre-change class (bounded, sub-Gaussian, Bernoulli or Poisson observations). Phase I/Phase II workflow, individual, p, c, u and regression charts, run-length simulation and auditable alarm reports.

License GPL (>= 3)

URL https://castlaboratory.github.io/edetect/, https://github.com/castlaboratory/edetect

BugReports https://github.com/castlaboratory/edetect/issues

Encoding UTF-8

Language en-GB

Depends R (>= 4.1)

Imports cli (>= 3.6.0), generics, ggplot2 (>= 3.5.0), rlang (>= 1.1.0), stats, tibble (>= 3.2.0), utils

Roxygen list(markdown = TRUE)

RoxygenNote 8.0.0

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

VignetteBuilder knitr

Config/testthat/edition 3

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

Date/Publication 2026-10-21 14:01:55 UTC

RemoteUrl https://github.com/castlaboratory/edetect

RemoteRef HEAD

RemoteSha 35c9b53808508404ced6db8364b2e0f566f91ce4

Contents

autoplot.edetect_chart . . . 2
edetect_alarm . . . 3
edetect_arl . . . 3
edetect_c . . . 4
edetect_classes . . . 5
edetect_design . . . 6
edetect_init . . . 7
edetect_p . . . 8
edetect_regression . . . 9
edetect_report . . . 9
edetect_restart . . . 10
edetect_u . . . 11
edetect_update . . . 11
edetect_x . . . 12
glance.edetect_chart . . . 13
tidy.edetect_chart . . . 13

Index **15**

`autoplot.edetect_chart`
Plot a detector

Description

Two panels: the observations (as values, proportions or rates per unit exposure) with the pre-change centre, and the evidence of the detector on the log10 scale with the alarm threshold and, if any, the alarm time and the estimated change point.

Usage

```
## S3 method for class 'edetect_chart'
autoplot(object, ...)
```

Arguments

`object`	An `edetect_chart`.
`...`	Unused.

Value

A ggplot.

Examples

```
chart <- edetect_c(x = c(3, 2, 4, 3, 8, 9, 11, 10), center = 3, arl = 100)
autoplot(chart)
```

edetect_alarm *The alarm status of a detector*

Description

The alarm status of a detector

Usage

```
edetect_alarm(state)
```

Arguments

state	An edetect_chart.

Value

A one-row tibble with alarm (logical), alarm_time, evidence at the alarm (or the current evidence), threshold, change_estimate (the start of the e-process that carries most of the current evidence) and n_observed.

Examples

```
design <- edetect_design(arl = 100, class = "subgaussian", center = 0, sigma = 1)
state <- edetect_update(edetect_init(design), rnorm(10))
edetect_alarm(state)
```

edetect_arl *Simulate run lengths of a design*

Description

Estimates the average run length to a false alarm (ARL, no change) or the detection delay (with a change) of an e-detector design by Monte Carlo. The recursion is vectorised across replications.

Usage

```
edetect_arl(
  design,
  pre,
  post = NULL,
  change_at = 1L,
  n_rep = 200L,
  max_t = NULL
)
```

Arguments

design	An edetect_design().
pre	A function function(n) returning n pre-change observations (for the "bernoulli" and "poisson" classes, a data frame with columns x and n).
post	Optional function with the same signature for post-change observations. When supplied, the change happens at time change_at.
change_at	Time of the first post-change observation.
n_rep	Number of replications.
max_t	Censoring time; runs that have not alarmed by max_t are reported as censored.

Value

A list of class edetect_runs with runs (a tibble with one row per replication: run_length, alarm, delay when there is a change, false_alarm) and summary (a one-row tibble with the mean run length or delay, its standard error, the censoring fraction and, with a change, the false-alarm fraction before change_at).

Examples

```
design <- edetect_design(arl = 50, class = "subgaussian", center = 0, sigma = 1)
set.seed(1)
edetect_arl(design, pre = function(n) rnorm(n), n_rep = 50, max_t = 2000)$summary
edetect_arl(design, pre = function(n) rnorm(n), post = function(n) rnorm(n, 1),
            change_at = 20, n_rep = 50, max_t = 500)$summary
```

---

edetect_c — *c chart: counts of nonconformities per unit*

Description

c chart: counts of nonconformities per unit

Usage

```
edetect_c(x, center = NULL, phase1 = NULL, ...)
```

Arguments

x	Counts per inspection unit.
center	Pre-change mean count; omit to estimate it from phase1.
phase1	Optional Phase I counts (a numeric vector, unit exposure).
...	Passed to edetect_design() (arl, direction, type, evalue, delta, weights).

Value

An edetect_chart.

Examples

```
chart <- edetect_c(x = c(3, 2, 4, 3, 8, 9, 11), center = 3, arl = 100)
edetect_alarm(chart)
```

---

edetect_classes *Pre-change classes*

Description

The classes of pre-change distributions for which `edetect` builds e-values. Each class is a statement about the observations *before* a change; the average run length guarantee holds for every distribution in the class.

Usage

```
edetect_classes()
```

Details

- `"bounded"`: observations in known bounds `[a, b]` with mean at most (direction `"up"`), at least (`"down"`) or equal to (`"both"`) `center`. Nothing else is assumed. E-values: `"betting"` (tight, default) or `"hoeffding"` (sub-Bernoulli).
- `"subgaussian"`: observations with mean `center` and sub-Gaussian tails with declared scale `sigma`. Gaussian data with standard deviation `sigma` is the boundary case; heavier tails violate the class.
- `"bernoulli"`: counts of successes in subgroups of size `n`, each trial with conditional success probability (given the earlier trials of the subgroup) at most/at least/equal to `center`; beyond that, dependence within a subgroup is allowed. This is the p chart.
- `"poisson"`: counts with exposure `n` whose moment generating function is at most that of a Poisson with rate `center` per unit exposure (Poisson and underdispersed counts; overdispersed counts violate the class). These are the c (`n = 1`) and u charts.

Value

A character vector of class names.

Examples

```
edetect_classes()
```

`edetect_design` *Design an e-detector*

Description

Fixes, before any Phase II observation is seen, everything the detector needs: the pre-change class and its parameters, the direction of the change to detect, the detector type (Shiryaev–Roberts or CUSUM), the family of e-values and the mixture over post-change alternatives, and the target average run length (ARL) to a false alarm.

Usage

```
edetect_design(
  arl = 370,
  class = edetect_classes(),
  center = NULL,
  bounds = NULL,
  sigma = NULL,
  direction = c("up", "down", "both"),
  type = c("sr", "cusum"),
  evalue = c("betting", "hoeffding"),
  delta = NULL,
  weights = NULL,
  phase1 = NULL
)
```

Arguments

`arl`	Target average run length to a false alarm. The alarm threshold is `arl` on the e-detector scale (equivalently `alpha = 1 / arl`).
`class`	Pre-change class, see `edetect_classes()`.
`center`	Pre-change mean (`"bounded"`, `"subgaussian"`), success probability (`"bernoulli"`) or rate per unit exposure (`"poisson"`). May be omitted when `phase1` is supplied.
`bounds`	Known bounds `c(a, b)` of the observations (`"bounded"` only).
`sigma`	Declared sub-Gaussian scale (`"subgaussian"` only). May be omitted when `phase1` is supplied, in which case the Phase I standard deviation is used and recorded as an assumption.
`direction`	Change to detect: an increase (`"up"`), a decrease (`"down"`) or either (`"both"`, a half-and-half mixture of the two).
`type`	`"sr"` (Shiryaev–Roberts type, a sum of e-processes started at every time) or `"cusum"` (CUSUM type, a maximum). Both are e-detectors and both use the threshold `arl`, which for the CUSUM type is valid but conservative (Shin, Ramdas and Rinaldo, 2024, Section 2.5); `"sr"` is the default.

`evalue`	E-value family for the `"bounded"` class: `"betting"` (`1 + lambda (x - center)`, tight for bounded data) or `"hoeffding"` (sub-Bernoulli exponential). Ignored by the other classes, which use their natural exponential e-values.
`delta`	Post-change shifts the mixture covers, in the units of the observations (positive numbers; the sign follows `direction`). Each shift is mapped to the bet `lambda` that is optimal for it in the class. The default is a geometric grid spanning small to large shifts.
`weights`	Mixture weights over `delta`, normalised internally. Default uniform.
`phase1`	Optional Phase I (in-control) data: a numeric vector for the `"bounded"` and `"subgaussian"` classes, or a data frame with columns `x` (counts) and `n` for `"bernoulli"` and `"poisson"`. Used to set `center` (and `sigma`) when they are not declared; the source is recorded in the design and in every report.

Details

The guarantee is distribution-free within the class: for every pre-change distribution in it, the expected time to a false alarm is at least `arl` (Shin, Ramdas and Rinaldo, 2024, Theorem 2.4), and the probability of a false alarm by any time `t`, including data-dependent monitoring horizons, is at most `t / arl` (Ramdas, 2026). The price is paid in detection delay, which depends on the alternatives the mixture covers.

Value

An object of class `edetect_design`.

References

Shin, J., Ramdas, A. and Rinaldo, A. (2024). E-detectors: a nonparametric framework for sequential change detection. *The New England Journal of Statistics in Data Science*, 2(2), 229–260. doi:10.51387/23NEJSDS51

Ramdas, A. (2026). Universality of e-detectors for ARL control. arXiv:2608.12660.

Examples

```
edetect_design(arl = 370, class = "bounded", center = 0.5, bounds = c(0, 1))
edetect_design(arl = 500, class = "bernoulli", center = 0.02, direction = "up")
edetect_design(arl = 370, class = "subgaussian", phase1 = rnorm(100), direction = "both")
```

`edetect_init` *Initialise an e-detector*

Description

Initialise an e-detector

Usage

```
edetect_init(design)
```

Arguments

design	An edetect_design().

Value

An object of class edetect_chart holding the design, the running state and an empty trajectory.

Examples

```
design <- edetect_design(arl = 370, class = "bounded", center = 0.5, bounds = c(0, 1))
state <- edetect_init(design)
state
```

---

edetect_p — *p chart: proportion of nonconforming units*

Description

p chart: proportion of nonconforming units

Usage

```
edetect_p(x, n, center = NULL, phase1 = NULL, ...)
```

Arguments

x	Counts of nonconforming units per subgroup.
n	Subgroup sizes (one per subgroup or a single value).
center	Pre-change proportion; omit to estimate it from phase1.
phase1	Optional Phase I data frame with columns x and n.
...	Passed to edetect_design() (arl, direction, type, evalue, delta, weights).

Value

An edetect_chart.

Examples

```
chart <- edetect_p(x = c(1, 0, 2, 1, 3, 4, 5, 6), n = 100, center = 0.01, arl = 100)
edetect_alarm(chart)
```

edetect_regression — *Regression chart: monitoring residuals of a Phase I model*

Description

Fits a linear model on Phase I data and monitors the Phase II residuals `y - prediction` as a sub-Gaussian stream centred at zero, with the residual standard deviation of the Phase I fit as the declared scale (or a `sigma` supplied by the user). The fitted model is kept in the chart.

Usage

```
edetect_regression(formula, phase1, phase2, sigma = NULL, ...)
```

Arguments

`formula`	Model formula.
`phase1`	Phase I data frame.
`phase2`	Phase II data frame with the same columns.
`sigma`	Declared sub-Gaussian scale of the residuals; default the Phase I residual standard deviation.
`...`	Passed to `edetect_design()` (`arl`, `direction`, `type`, `evalue`, `delta`, `weights`).

Value

An `edetect_chart` with an extra element `fit`.

Examples

```
set.seed(3)
d1 <- data.frame(z = 1:60); d1$y <- 2 + 0.5 * d1$z + rnorm(60, sd = 0.3)
d2 <- data.frame(z = 61:100)
d2$y <- 2 + 0.5 * d2$z + rnorm(40, sd = 0.3) + c(rep(0, 20), rep(1, 20))
chart <- edetect_regression(y ~ z, phase1 = d1, phase2 = d2, arl = 200)
edetect_alarm(chart)
```

edetect_report — *The full report of a detector*

Description

Everything a reader needs to audit an alarm or its absence: what was assumed (class, centre and its source, bounds or scale, direction), how the detector was built (type, e-value family, mixture), what was observed, the outcome and the software versions.

Usage

```
edetect_report(state)
```

Arguments

state	An edetect_chart.

Value

A list of class edetect_report.

Examples

```
design <- edetect_design(arl = 100, class = "bernoulli", center = 0.05)
state <- edetect_update(edetect_init(design), x = c(2, 3, 1, 4), n = 50)
edetect_report(state)
```

---

edetect_restart *Restart a detector after an alarm*

Description

Returns a fresh detector with the same design. The history of the previous run is kept in the runs element of the new object, so that a report can list every alarm.

Usage

```
edetect_restart(state)
```

Arguments

state	An edetect_chart that raised an alarm.

Value

A new edetect_chart.

Examples

```
design <- edetect_design(arl = 20, class = "subgaussian", center = 0, sigma = 1)
state <- suppressWarnings(edetect_update(edetect_init(design), c(rnorm(5), rnorm(30, 3))))
edetect_alarm(state)$alarm
state <- edetect_restart(state)
edetect_alarm(state)$alarm
```

edetect_u — *u chart: nonconformities per unit of exposure*

Description

u chart: nonconformities per unit of exposure

Usage

```
edetect_u(x, n, center = NULL, phase1 = NULL, ...)
```

Arguments

x	Counts per sample.
n	Exposure (area of opportunity) of each sample.
center	Pre-change rate per unit exposure; omit to estimate it from phase1.
phase1	Optional Phase I data frame with columns x and n.
...	Passed to edetect_design() (arl, direction, type, evalue, delta, weights).

Value

An edetect_chart.

Examples

```
chart <- edetect_u(x = c(3, 2, 4, 3, 8, 9, 11), n = c(2, 2, 2, 2, 2, 2, 2), center = 1.5, arl = 100)
edetect_alarm(chart)
```

edetect_update — *Update an e-detector with new observations*

Description

Feeds one or more Phase II observations, in time order, to the detector. The first time the statistic reaches the threshold an alarm is recorded; the detector does not accept further observations after that (see edetect_restart()).

Usage

```
edetect_update(state, x, n = NULL)
```

Arguments

state	An edetect_chart from edetect_init().
x	Numeric vector of observations (values for the "bounded" and "subgaussian" classes; counts for "bernoulli" and "poisson").
n	Subgroup sizes ("bernoulli") or exposures ("poisson"), one per observation or a single value. Ignored by the other classes.

Value

The updated edetect_chart.

Examples

```
design <- edetect_design(arl = 100, class = "subgaussian", center = 0, sigma = 1)
state <- edetect_init(design)
set.seed(1)
state <- edetect_update(state, c(rnorm(30), rnorm(30, mean = 1.5)))
edetect_alarm(state)
```

---

edetect_x *Individual-observation chart for bounded data*

Description

Individual-observation chart for bounded data

Usage

```
edetect_x(x, bounds, center = NULL, phase1 = NULL, ...)
```

Arguments

x	Phase II observations.
bounds	Known bounds of the observations.
center	Pre-change mean; omit to estimate it from phase1.
phase1	Optional Phase I observations.
...	Passed to edetect_design() (arl, direction, type, evalue, delta, weights).

Value

An edetect_chart.

Examples

```
set.seed(2)
x <- c(runif(40, 0.3, 0.7), runif(40, 0.5, 0.9))
chart <- edetect_x(x, bounds = c(0, 1), center = 0.5, arl = 200)
edetect_alarm(chart)
```

glance.edetect_chart *Glance at a detector*

Description

Glance at a detector

Usage

```
## S3 method for class 'edetect_chart'
glance(x, ...)
```

Arguments

x	An edetect_chart.
...	Unused.

Value

A one-row tibble: class, center, center_source, direction, type, arl, n_observed, alarm, alarm_time, evidence, change_estimate.

Examples

```
chart <- edetect_c(x = c(3, 2, 4, 3, 8, 9, 11), center = 3, arl = 100)
glance(chart)
```

tidy.edetect_chart *Tidy the trajectory of a detector*

Description

One row per observation with the mixed log e-value, the detector evidence (SR or CUSUM type, as designed), the CUSUM-type evidence, the threshold, the running change-point estimate and the alarm flag.

Usage

```
## S3 method for class 'edetect_chart'
tidy(x, ...)
```

Arguments

x	An edetect_chart.
...	Unused.

Value

A tibble with columns `t`, `x`, `n`, `log_evalue`, `evidence`, `evidence_cusum`, `threshold`, `change_estimate`, `alarm`.

Examples

```
chart <- edetect_c(x = c(3, 2, 4, 3, 8, 9, 11), center = 3, arl = 100)
tidy(chart)
```

Index

autoplot.edetect_chart, 2

edetect_alarm, 3
edetect_arl, 3
edetect_c, 4
edetect_classes, 5
edetect_classes(), *6*
edetect_design, 6
edetect_design(), *4*, *8*, *9*, *11*, *12*
edetect_init, 7
edetect_init(), *12*
edetect_p, 8
edetect_regression, 9
edetect_report, 9
edetect_restart, 10
edetect_restart(), *11*
edetect_u, 11
edetect_update, 11
edetect_x, 12

glance.edetect_chart, 13

tidy.edetect_chart, 13