

Package ‘deriva’

September 20, 2026

Title Tidy Drift Detection for Monitored Machine Learning Models

Version 0.2.0

Description Detects concept drift and data drift in streams produced by deployed machine learning models, using a tidy interface that composes with the 'tidymodels' ecosystem. Detectors are specified, fitted on a baseline period, and advanced over new batches of observations, returning tibbles annotated with warning and drift flags. A catalogue of 22 sequential drift detectors is provided. Error-based methods include the Drift Detection Method (DDM) of Gama et al. (2004) [<doi:10.1007/978-3-540-28645-5_29>](https://doi.org/10.1007/978-3-540-28645-5_29), the Early Drift Detection Method (EDDM) of Baena-Garcia et al. (2006), the Hoeffding's inequality based Drift Detection Methods (HDDM) of Frias-Blanco et al. (2015) [<doi:10.1109/TKDE.2014.2345382>](https://doi.org/10.1109/TKDE.2014.2345382), and the Exponentially Weighted Moving Average (EWMA) chart of Ross et al. (2012) [<doi:10.1016/j.patrec.2011.08.019>](https://doi.org/10.1016/j.patrec.2011.08.019). Distribution-based methods include Adaptive Windowing (ADWIN) of Bifet and Gavalda (2007) [<doi:10.1137/1.9781611972771.42>](https://doi.org/10.1137/1.9781611972771.42), Kolmogorov-Smirnov Windowing (KSWIN) of Raab et al. (2020) [<doi:10.1016/j.neucom.2019.11.111>](https://doi.org/10.1016/j.neucom.2019.11.111), and the Page-Hinkley test of Page (1954) [<doi:10.1093/biomet/41.1-2.100>](https://doi.org/10.1093/biomet/41.1-2.100).

License MIT + file LICENSE

URL <https://github.com/bonijoao/deriva>,
<https://bonijoao.github.io/deriva/>

BugReports <https://github.com/bonijoao/deriva/issues>

Encoding UTF-8

LazyData true

RoxygenNote 7.3.3

Depends R (>= 4.1)

Imports cli, generics, rlang, stats, tibble, utils, vctrs, withr

Suggests ggplot2, knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation no

Author João Paulo Assis Bonifácio [aut, cre] (ORCID:
<<https://orcid.org/0009-0009-2335-8391>>),
Geraldo Magela da Cruz Pereira [aut] (ORCID:
<<https://orcid.org/0000-0001-6280-4870>>),
Pedro Mambelli Fernandes [aut] (ORCID:
<<https://orcid.org/0009-0001-1695-279X>>)

Maintainer João Paulo Assis Bonifácio <jpab.27@hotmail.com>

Repository CRAN

Date/Publication 2026-09-20 17:50:02 UTC

Contents

add_prediction_error	2
advance	3
augment.drift_detector_fit	4
autoplot.drift_detector_fit	5
credit_monitoring	5
detect_drift	6
drift_detector	7
fit.drift_detector	8
glance.drift_detector_fit	9
sensor_monitoring	9
sim_dist_stream	10
sim_drift_stream	11
tidy.drift_detector_fit	11
Index	13

add_prediction_error	<i>Build a drift signal from model predictions</i>
----------------------	--

Description

Bridge from tidymodels: takes the output of augment() on a fitted workflow/model and adds a .error column — the signal drift detectors consume. Classification (factor/character truth): 0/1 mismatch against estimate (default column .pred_class). Regression (numeric truth): absolute error against estimate (default column .pred).

Usage

add_prediction_error(data, truth, estimate = NULL, ...)

Arguments

<code>data</code>	A data frame with truth and prediction columns.
<code>truth</code>	Unquoted name of the true outcome column.
<code>estimate</code>	Unquoted name of the prediction column. Defaults to <code>.pred_class</code> (classification) or <code>.pred</code> (regression), following tidymodels conventions.
<code>...</code>	Not used.

Value

data as a tibble with a `.error` column added.

Examples

```
d <- tibble::tibble(truth = c(1, 2, 3), .pred = c(1, 1, 5))
add_prediction_error(d, truth = truth)
```

advance

Advance a fitted drift detector over a new batch

Description

Feeds a new batch of observations (any size, including 1 — stream mode) to the detector and returns a NEW fitted object with the engine state advanced and the annotated batch appended to the history (truncated to the last keep rows of the spec). The original object is not modified. This is the only way to persist state; see [augment\(\)](#) for a read-only preview.

Usage

```
advance(object, ...)
```

```
## S3 method for class 'drift_detector_fit'
advance(object, new_data, ...)
```

Arguments

<code>object</code>	A <code>drift_detector_fit</code> .
<code>...</code>	Passed to methods.
<code>new_data</code>	A data frame with the new batch, in temporal order, containing the same signal column used in fit() . It must have the same columns as the data given to fit() ; deriva's own columns (<code>.warning</code> , <code>.drift</code> , <code>.phase</code>) must not be present.

Details

Why not `update()`: in the tidymodels ecosystem `update()` on a spec means "change hyperparameters", so deriva defines its own verb.

Value

A new drift_detector_fit.

Examples

```
base <- sim_drift_stream(n_pre = 100, n_post = 0, seed = 1)
f0 <- fit(drift_detector("ddm"), base, signal = error)
f1 <- advance(f0, sim_drift_stream(n_pre = 0, n_post = 50, seed = 2))
```

augment.drift_detector_fit

Annotated observations from a fitted drift detector

Description

With new_data = NULL, returns the retained history (the last keep rows of baseline + advanced batches) annotated with .warning, .drift and .phase. With new_data, returns a READ-ONLY preview: the batch annotated from the current state, WITHOUT persisting it — use [advance\(\)](#) to persist.

Usage

```
## S3 method for class 'drift_detector_fit'
augment(x, new_data = NULL, ...)
```

Arguments

x	A drift_detector_fit.
new_data	Optional data frame with a new batch to preview.
...	Not used.

Value

A tibble.

Examples

```
base <- sim_drift_stream(n_pre = 100, n_post = 0, seed = 1)
f0 <- fit(drift_detector("ddm"), base, signal = error)
augment(f0)
```

```
autoplot.drift_detector_fit
```

Plot the monitored signal with drift markings

Description

Plots the running mean of the signal over the retained history (see keep in `drift_detector()`), with the baseline/stream boundary (labelled "training ends"), warning points (orange) and drift points (red vertical lines; the first drift within the retained history is labelled with its index). Requires ggplot2 (Suggests).

Usage

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

Arguments

<code>object</code>	A <code>drift_detector_fit</code> .
<code>...</code>	Not used.

Value

A ggplot object.

<code>credit_monitoring</code>	<i>Simulated credit monitoring stream</i>
--------------------------------	---

Description

A synthetic per-observation error stream from a credit-approval classifier in production: 500 stable observations (5% error rate), then 500 after a market shift raised the error rate to 30%. Frozen with a fixed seed, so every example that loads it sees the same story — including the single, correct DDM detection at $t = 542$ with no false positives before it.

Usage

```
credit_monitoring
```

Format

A tibble with 1,000 rows and 3 columns:

t Observation index, 1 to 1000.

error 0/1 classifier error for that observation.

drift_true Ground truth: TRUE from observation 501 on, the point the market shift occurred.

Source

Simulated with `sim_drift_stream()`: `sim_drift_stream(n_pre = 500, n_post = 500, p_pre = 0.05, p_post = 0.30, seed = 2)`. See `data-raw/credit-monitoring.R`.

Examples

```
credit_monitoring
detect_drift(credit_monitoring, .col = error, method = "ddm")
```

detect_drift	<i>Detect drift in a signal column (one-shot shortcut)</i>
--------------	--

Description

Layer-3 convenience: runs a detector over an existing signal column and returns the data annotated with `.warning` / `.drift`. For an explicit baseline and persistent state, use the full object path: `drift_detector() + fit() + advance()`.

Usage

```
detect_drift(data, .col, method = "ddm", ...)
```

Arguments

<code>data</code>	A data frame in temporal order.
<code>.col</code>	Unquoted name of the signal column.
<code>method</code>	Name of a registered method (default "ddm").
<code>...</code>	Hyperparameters and seed, forwarded to <code>drift_detector()</code> .

Details

For "ddm", the warm-up is governed by `min_instances`: the first `min_instances - 1` observations get NA flags.

Value

data as a tibble with `.warning` and `.drift` columns added.

Examples

```
s <- sim_drift_stream(seed = 42)
detect_drift(s, .col = error, method = "ddm")
```

drift_detector	<i>Specify a drift detector</i>
----------------	---------------------------------

Description

Creates an inert detector specification (analogous to a parsnip model spec). Nothing is computed until `fit()` is called on a baseline period.

Usage

```
drift_detector(method = "ddm", ..., seed = NULL, keep = 10000)
```

Arguments

method	Name of a registered detection method, e.g. "ddm".
...	Method hyperparameters overriding the defaults (e.g. <code>min_instances = 50</code> for "ddm"). Unknown parameters and values outside a parameter's valid range error. A few hyperparameters are thresholds on the signal's own scale rather than dimensionless, notably <code>epsilon_prime</code> for "seed", whose default (0.01) suits a 0/1 error stream; on a numeric stream of a different magnitude, scale it to match.
seed	NULL (default) or a single whole number. When set, the detector draws from its own private random stream, carried inside the fitted object: results are reproducible, do not depend on how the stream is split into batches, and the session's global RNG is left untouched. Only "kswin" and "seqdrift2" are stochastic. With NULL they draw from the global RNG, so call <code>set.seed()</code> yourself for reproducibility.
keep	Number of most recent annotated rows retained in the fitted object's history (default 10000), bounding memory and the cost of each <code>advance()</code> on long-running streams. Totals in <code>glance()</code> and drift points in <code>tidy()</code> are tracked separately and stay exact whatever keep is. Inf keeps everything and 0 keeps nothing; both warn.

Value

A `drift_detector` specification object.

Warning and drift flags

Every detector annotates each observation with `.warning` and `.drift` under one contract. NA: the detector cannot judge this observation yet — it is warming up, which also happens again right after a detected drift resets it. FALSE: the detector is active and has not flagged drift as of this observation; note that "adwin", "seed" and "seqdrift2" run their test only on a clock or at block boundaries, so between tests they carry the previous verdict forward. TRUE: it flagged drift here. Detectors with no warning level ("ewma", "page_hinkley", "cusum", "kswin", "adwin", "seed", "seqdrift2", "fhddms", "mddm_a", "mddm_g", "mddm_e") always give `.warning = NA`. Use `which(.drift)` or `dplyr::filter(.drift)`, which skip NA; `any(.drift)` needs `na.rm = TRUE`.

Examples

```
drift_detector("ddm", min_instances = 50)
```

fit.drift_detector	<i>Fit a drift detector on a baseline period</i>
--------------------	--

Description

Runs the detector over the baseline data — the period where the monitored model is considered stable — so it learns the reference ("normal") level. The returned object is immutable: feed new batches with [advance\(\)](#).

Usage

```
## S3 method for class 'drift_detector'
fit(object, data, signal, ...)
```

Arguments

object	A drift_detector() specification.
data	A data frame with the baseline period, in temporal order.
signal	Unquoted name of the signal column (0/1 errors for error-based methods such as "ddm").
...	Not used.

Value

A drift_detector_fit object.

Examples

```
base <- sim_drift_stream(n_pre = 100, n_post = 0, seed = 1)
fit(drift_detector("ddm"), base, signal = error)
```

glance.drift_detector_fit

One-row summary of a fitted detector

Description

One-row summary of a fitted detector

Usage

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

Arguments

x A drift_detector_fit.
... Not used.

Value

A 1-row tibble: method, n_obs, n_warning, n_drift, first_drift (NA if no drift detected).

sensor_monitoring

Simulated sensor monitoring stream

Description

A synthetic numeric sensor-reading stream: 500 stable observations centred at 0, then 500 after the sensor drifted out of calibration and the mean shifted to 2. Frozen with a fixed seed.

Usage

```
sensor_monitoring
```

Format

A tibble with 1,000 rows and 3 columns:

t Observation index, 1 to 1000.

value Numeric sensor reading.

drift_true Ground truth: TRUE from observation 501 on, the point the sensor drifted.

Source

Simulated with `sim_dist_stream()`: `sim_dist_stream(n_pre = 500, n_post = 500, mean_pre = 0, mean_post = 2, seed = 2)`. See `data-raw/sensor-monitoring.R`.

Examples

```
sensor_monitoring
detect_drift(sensor_monitoring, .col = value, method = "kswin", seed = 7)
```

sim_dist_stream

Simulate a continuous stream with a known distribution-shift point

Description

Generates a numeric stream drawn from $N(\text{mean_pre}, \text{sd_pre})$ for the first n_pre observations and $N(\text{mean_post}, \text{sd_post})$ afterwards. Companion to [sim_drift_stream\(\)](#) for distribution-based detectors (e.g. "kswin").

Usage

```
sim_dist_stream(
  n_pre = 500,
  n_post = 500,
  mean_pre = 0,
  mean_post = 3,
  sd_pre = 1,
  sd_post = 1,
  seed = NULL
)
```

Arguments

`n_pre, n_post` Observations before / after the shift point.
`mean_pre, mean_post` Means before / after the shift.
`sd_pre, sd_post` Standard deviations before / after the shift.
`seed` Optional integer for `set.seed()`.

Value

A tibble with `t` (index), `value` (numeric) and `drift_true` (logical: TRUE after the shift point).

Examples

```
sim_dist_stream(n_pre = 100, n_post = 100, mean_post = 3, seed = 42)
```

sim_drift_stream	<i>Simulate a binary error stream with a known drift point</i>
------------------	--

Description

Generates a stream of 0/1 classifier errors whose error rate jumps from `p_pre` to `p_post` after `n_pre` observations. Useful for testing and validating drift detectors against a known ground truth.

Usage

```
sim_drift_stream(
  n_pre = 500,
  n_post = 500,
  p_pre = 0.05,
  p_post = 0.3,
  seed = NULL
)
```

Arguments

<code>n_pre</code> , <code>n_post</code>	Number of observations before / after the drift point.
<code>p_pre</code> , <code>p_post</code>	Error probability before / after the drift point.
<code>seed</code>	Optional integer; if supplied, <code>set.seed()</code> is called for reproducibility.

Value

A tibble with columns `t` (index), `error` (0/1) and `drift_true` (logical ground truth: TRUE after the drift point).

Examples

```
sim_drift_stream(n_pre = 100, n_post = 100, seed = 42)
```

tidy.drift_detector_fit	<i>Drift points of a fitted detector</i>
-------------------------	--

Description

Drift points of a fitted detector

Usage

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

Arguments

x	A <code>drift_detector_fit</code> .
...	Not used.

Value

A tibble with one row per detected drift: `index` (position since the start of the baseline, exact even when the history is truncated by `keep`) and `phase`.

Index

* datasets

- credit_monitoring, 5
- sensor_monitoring, 9

- add_prediction_error, 2
- advance, 3
- advance(), 4, 6–8
- augment(), 3
- augment.drift_detector_fit, 4
- autoplot.drift_detector_fit, 5

- credit_monitoring, 5

- detect_drift, 6
- drift_detector, 7
- drift_detector(), 5, 6, 8

- fit(), 3, 6, 7
- fit.drift_detector, 8

- glance(), 7
- glance.drift_detector_fit, 9

- sensor_monitoring, 9
- set.seed(), 7
- sim_dist_stream, 10
- sim_dist_stream(), 9
- sim_drift_stream, 11
- sim_drift_stream(), 6, 10

- tidy(), 7
- tidy.drift_detector_fit, 11