

Package: HLCtools (via r-universe)

July 23, 2026

Title Calculate Herd Lying Concordance Metrics

Version 0.1.0

Description Calculates Herd Lying Concordance (HLC) metrics from individual animal lying-behaviour data. HLC is a continuous framework for quantifying group-level behavioural cohesion from between-animal dispersion within observation intervals. The package implements standard deviation, mean absolute deviation, interquartile range, and entropy formulations, lying-weighted extensions, threshold-based synchrony comparisons, temporal summaries, and descriptive method-ranking tools. The HLC formulations are introduced in this package; related approaches to cattle behavioural synchrony include Raussi et al. (2011) <[doi:10.1017/S1751731110001928](https://doi.org/10.1017/S1751731110001928)>, Kok et al. (2023) <[doi:10.1016/j.applanim.2023.105906](https://doi.org/10.1016/j.applanim.2023.105906)>, and the activity metric framework of van Dixhoorn et al. (2024) <[doi:10.24072/pcjournal.489](https://doi.org/10.24072/pcjournal.489)>.

License MIT + file LICENSE

URL <https://github.com/guilhermefranchi/HLCtools>

BugReports <https://github.com/guilhermefranchi/HLCtools/issues>

Encoding UTF-8

Language en-US

RoxygenNote 7.3.3

Depends R (>= 4.1.0)

Imports dplyr, rlang

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

LazyData true

NeedsCompilation no

Author Guilherme Amorim Franchi [aut, cre, cph] (ORCID: <<https://orcid.org/0000-0001-7653-6512>>)

Maintainer Guilherme Amorim Franchi <gaf@anivet.au.dk>
Repository <https://cran.r-universe.dev>
Date/Publication 2026-07-23 13:10:15 UTC
RemoteUrl <https://github.com/cran/HLCtools>
RemoteRef HEAD
RemoteSha 4f78d8e267311f9bc3b3e2084518fba87da15540

Contents

calculate_hlc	2
compare_hlc_methods	4
example_lies	5
hlc_methods	6
hlc_weighted_methods	6
rank_hlc_methods	7
summarise_hlc_daily	9

Index	10
--------------	-----------

calculate_hlc	<i>Calculate interval-level Herd Lying Concordance</i>
---------------	--

Description

Calculates Herd Lying Concordance (HLC) metrics from individual animal lying-time records. HLC is calculated within each group-time interval as one minus a normalised measure of between-animal dispersion.

Usage

```
calculate_hlc(  
  data,  
  group,  
  animal,  
  interval,  
  lying,  
  day = NULL,  
  period = NULL,  
  interval_min = 15,  
  methods = c("sd", "mad", "iqr", "entropy"),  
  lying_threshold = interval_min/2,  
  sync_thresholds = c(0.6, 0.7, 0.8, 0.9),  
  add_lying_weighted = TRUE  
)
```

Arguments

<code>data</code>	A data frame containing individual animal behaviour records.
<code>group</code>	Column identifying the herd, treatment, pen, or group.
<code>animal</code>	Column identifying the individual animal.
<code>interval</code>	Column identifying the time interval within day.
<code>lying</code>	Column containing lying time within the interval, in minutes.
<code>day</code>	Optional column identifying day/date.
<code>period</code>	Optional column identifying a larger period, for example week.
<code>interval_min</code>	Length of the observation interval in minutes. Default is 15.
<code>methods</code>	Character vector of HLC methods to compute. Options are "sd", "mad", "iqr", and "entropy".
<code>lying_threshold</code>	Threshold in minutes used to define binary lying state for entropy. Default is half of <code>interval_min</code> .
<code>sync_thresholds</code>	Numeric vector of threshold synchrony cut-offs, expressed as proportions. Default is <code>c(0.6, 0.7, 0.8, 0.9)</code> .
<code>add_lying_weighted</code>	Logical. If TRUE, adds lying-weighted HLC metrics.

Value

A tibble with one row per group-time interval.

Examples

```
data(example_lies)

hlc_intervals <- calculate_hlc(
  data = example_lies,
  group = group,
  animal = cow,
  day = day,
  period = week,
  interval = time,
  lying = lying,
  interval_min = 15,
  methods = c("sd", "mad", "iqr", "entropy"),
  sync_thresholds = c(0.6, 0.7, 0.8, 0.9),
  add_lying_weighted = TRUE
)

hlc_intervals
```

compare_hlc_methods	<i>Compare HLC methods</i>
---------------------	----------------------------

Description

Summarises HLC metrics across available dispersion bases. This function helps users compare HLC implementations rather than automatically selecting a single best metric.

Usage

```
compare_hlc_methods(data, metrics = NULL)
```

Arguments

data	A data frame containing HLC summary columns. This can be daily, weekly, treatment-level, or another summarised object.
metrics	Optional character vector of HLC columns to compare. If NULL, columns matching mean_HLC_SD, mean_HLC_MAD, mean_HLC_IQR, mean_HLC_ENT, HLC_SD, HLC_MAD, HLC_IQR, and HLC_ENT are used when available.

Value

A tibble with descriptive summaries for each available HLC method.

Examples

```
data(example_lies)

hlc_intervals <- calculate_hlc(
  data = example_lies,
  group = group,
  animal = cow,
  day = day,
  period = week,
  interval = time,
  lying = lying
)

hlc_daily <- summarise_hlc_daily(
  data = hlc_intervals,
  group = group,
  day = day,
  period = week
)

compare_hlc_methods(hlc_daily)
```

example_lies	<i>Example lying behaviour dataset</i>
--------------	--

Description

A small synthetic dataset for demonstrating Herd Lying Concordance calculations. The dataset contains individual animal lying times for two groups, two days, two time intervals per day, and five cows per group-time interval.

Usage

```
example_lies
```

Format

A data frame with 40 rows and 6 columns:

group Group identifier.

day Day identifier.

week Week identifier.

time Time interval identifier.

cow Individual cow identifier.

lying Lying time within the interval, in minutes.

Examples

```
data(example_lies)

hlc_intervals <- calculate_hlc(
  data = example_lies,
  group = group,
  animal = cow,
  day = day,
  period = week,
  interval = time,
  lying = lying
)

hlc_intervals
```

hlc_methods	<i>List available unweighted HLC methods</i>
-------------	--

Description

Returns the currently implemented unweighted Herd Lying Concordance dispersion bases.

Usage

```
hlc_methods()
```

Value

A character vector of available unweighted method names.

hlc_weighted_methods	<i>List available lying-weighted HLC methods</i>
----------------------	--

Description

Returns the names of the currently implemented lying-weighted Herd Lying Concordance methods.

Usage

```
hlc_weighted_methods()
```

Value

A character vector containing the available lying-weighted HLC method names.

Examples

```
hlc_weighted_methods()
```

rank_hlc_methods	<i>Rank HLC methods for a dataset</i>
------------------	---------------------------------------

Description

Ranks available Herd Lying Concordance (HLC) implementations using descriptive criteria relevant for method selection. The function supports daily, weekly, treatment-level, or other summarised HLC outputs.

Usage

```
rank_hlc_methods(
  data,
  group = NULL,
  period = NULL,
  lying_prop = NULL,
  metrics = NULL,
  weights = c(detectability = 1, temporal_stability = 1, outlier_robustness = 1, boundary
    = 1, missingness = 1, degeneracy = 1)
)
```

Arguments

data	A data frame containing HLC summary columns.
group	Optional column identifying treatment, herd, pen, farm, or another comparison group. If supplied, group detectability is evaluated.
period	Optional column identifying week, period, block, or another temporal unit. If supplied, temporal stability is evaluated.
lying_prop	Optional column containing lying proportion. If supplied, Spearman correlation between each HLC method and lying proportion is reported.
metrics	Optional character vector of HLC columns to rank. If NULL, columns matching mean_HLC_SD, mean_HLC_MAD, mean_HLC_IQR, mean_HLC_ENT, HLC_SD, HLC_MAD, HLC_IQR, and HLC_ENT are used when available.
weights	Named numeric vector giving weights for ranking criteria. Supported names are "detectability", "temporal_stability", "outlier_robustness", "boundary", "missingness", and "degeneracy".

Details

The ranking is intended to support transparent, dataset-specific selection of an HLC implementation. It does not imply that one dispersion basis is universally best.

Ranking criteria:

- **detectability:** group separation, based on an F statistic from nested linear models when group is supplied.

- temporal_stability: SD of period-level means when period is supplied.
- outlier_robustness: absolute distance of SD/MAD from 1.
- boundary: percentage of exact 0 or 1 values.
- missingness: percentage of missing values.
- degeneracy: penalises metrics with zero variance or very high boundary collapse.

Criteria that cannot be evaluated because the required columns were not supplied are omitted from the weighted score. Metrics with missing values for an otherwise available criterion are ranked last for that criterion.

Value

A tibble summarising method performance and ranking. Lower weighted_rank_score indicates a more favourable method under the chosen criteria. The column selected marks the top-ranked method.

Examples

```
data(example_lies)

hlc_intervals <- calculate_hlc(
  data = example_lies,
  group = group,
  animal = cow,
  day = day,
  period = week,
  interval = time,
  lying = lying
)

hlc_daily <- summarise_hlc_daily(
  data = hlc_intervals,
  group = group,
  day = day,
  period = week
)

rank_hlc_methods(
  data = hlc_daily,
  group = group,
  period = week,
  lying_prop = mean_lying_prop
)
```

summarise_hlc_daily	<i>Summarise HLC metrics by day</i>
---------------------	-------------------------------------

Description

Aggregates interval-level HLC metrics into daily group-level summaries.

Usage

```
summarise_hlc_daily(data, group, day, period = NULL, interval_min = 15)
```

Arguments

data	Interval-level HLC data produced by <code>calculate_hlc()</code> .
group	Column identifying herd, treatment, pen, or group.
day	Column identifying day/date.
period	Optional column identifying a larger period, for example week.
interval_min	Length of the observation interval in minutes. Default is 15.

Value

A tibble with daily HLC summaries.

Examples

```
data(example_lies)

hlc_intervals <- calculate_hlc(
  data = example_lies,
  group = group,
  animal = cow,
  day = day,
  period = week,
  interval = time,
  lying = lying,
  interval_min = 15
)

hlc_daily <- summarise_hlc_daily(
  data = hlc_intervals,
  group = group,
  day = day,
  period = week,
  interval_min = 15
)

hlc_daily
```

Index

* **datasets**

example_lies, [5](#)

calculate_hlc, [2](#)

compare_hlc_methods, [4](#)

example_lies, [5](#)

hlc_methods, [6](#)

hlc_weighted_methods, [6](#)

rank_hlc_methods, [7](#)

summarise_hlc_daily, [9](#)