Package ‘PSRICalcSM’

May 7, 2026

Type Package

Title Plant Stress Response Index Calculator - Softmax Method

Version 1.0.0

Description Implements the softmax aggregation method for calculating Plant Stress Response Index (PSRI) from time-series germination data under environmental stressors including prions, xenobiotics, osmotic stress, heavy metals, and chemical contaminants. Provides zero-robust PSRI computation through adaptive softmax weighting of germination components (Maximum Stress-adjusted Germination, Maximum Rate of Germination, complementary Mean Time to Germination, and Radicle Vigor Score), eliminating the zero-collapse failure mode of the geometric mean approach implemented in 'PSRICalc'. Includes perplexity-based temperature parameter calibration and modular component functions for transparent germination analysis. Built on the methodological foundation of the Osmotic Stress Response Index (OSRI) framework developed by Walne et al. (2020) <doi:10.1002/agg2.20087>. Note: This package implements methodology currently under peer review. Please contact the author before publication using this approach.

Development followed an iterative human-machine collaboration where all algorithmic design, statistical methodologies, and biological validation logic were conceptualized, tested, and iteratively refined by Richard A. Feiss through repeated cycles of running experimental data, evaluating analytical outputs, and selecting among candidate algorithms and approaches. AI systems (Anthropic Claude and OpenAI GPT) served as coding assistants and analytical sounding boards under continuous human direction. The selection of statistical methods, evaluation of biological plausibility, and all final methodology decisions were made by the human author. AI systems did not independently originate algorithms, statistical approaches, or scientific methodologies.

Depends R (>= 4.0.0)

Suggests testthat (>= 3.0.0)

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

URL https://github.com/RFeissIV/PSRICalcSM

BugReports https://github.com/RFeissIV/PSRICalcSM/issues

Config/testthat/edition 3

NeedsCompilation no

Author Richard Feiss [aut, cre] (ORCID: <https://orcid.org/0009-0008-0409-6042>), University of Minnesota [cph]

Maintainer Richard Feiss <feiss026@umn.edu>

Repository CRAN

Date/Publication 2026-02-20 11:30:07 UTC

Contents

`PSRICalcSM-package` *Plant Stress Response Index Calculator - Softmax Method*

Description

Implements the softmax aggregation method for calculating Plant Stress Response Index (PSRI) from time-series germination data. Provides a zero-robust alternative to the geometric mean PSRI in **PSRICalc**.

Details

The main function is `compute_psri_sm`, which computes the softmax PSRI from germination count data. Component functions (`compute_msg`, `compute_mrg`, `compute_cmtg`, `compute_rvs`) are available for modular analysis. Use `calibrate_temperature` for data-driven temperature parameter selection.

Author(s)

Richard A. Feiss IV

Maintainer: Richard A. Feiss IV <feiss026@umn.edu>

References

Walne, C.H., Gaudin, A., Henry, W.B., and Reddy, K.R. (2020). In vitro seed germination response of corn hybrids to osmotic stress conditions. *Agrosystems, Geosciences & Environment*, 3(1), e20087. doi:10.1002/agg2.20087

See Also

`compute_psri_sm`, `calibrate_temperature`

`calibrate_temperature` *Calibrate Softmax Temperature Parameter*

Description

Selects the optimal temperature parameter for softmax PSRI aggregation using perplexity targeting. The temperature controls how sharply the softmax concentrates weight on the dominant component(s).

Usage

```
calibrate_temperature(
  component_profiles,
  target_perplexity = 2.0,
  T_range = c(0.01, 2.0),
  tolerance = 0.05
)
```

Arguments

`component_profiles`	A list of numeric vectors, where each vector represents a set of PSRI component scores from representative replicates.
`target_perplexity`	Numeric. Target effective number of components. Default is 2.0.
`T_range`	Numeric vector of length 2. Search range for T. Default is `c(0.01, 2.0)`.
`tolerance`	Numeric. Convergence tolerance. Default is 0.05.

Details

Perplexity measures effective components receiving meaningful weight:

$$Perplexity = \exp\left(-\sum_i W_i \log W_i\right)$$

A perplexity of 2.0 (default target, for 3 components) balances signal extraction with robustness. The calibration uses bisection search over the provided range.

Value

A list with:

optimal_T	The calibrated temperature parameter.
mean_perplexity	Mean perplexity at the optimal T.
profile_perplexities	Per-profile perplexity values.
target_perplexity	The specified target.
search_range	The search range used.

See Also

`softmax_weights`, `compute_psri_sm`

Examples

```
profiles <- list(
  corn_control   = c(0.80, 0.90, 0.60),
  corn_treated   = c(0.50, 0.40, 0.55),
  barley_control = c(0.35, 0.30, 0.45),
  barley_treated = c(0.20, 0.15, 0.50)
)
cal <- calibrate_temperature(profiles)
cal$optimal_T
cal$mean_perplexity
```

`compute_cmtg` *Compute Complementary Mean Time to Germination (cMTG)*

Description

Calculates cMTG as the complement of the normalized time to 50% germination, bounded in `[0, 1]`.

Usage

```
compute_cmtg(germination_counts, timepoints, total_seeds)
```

Arguments

`germination_counts`	Integer vector of cumulative germination counts at each timepoint.
`timepoints`	Numeric vector of observation times.
`total_seeds`	Integer. Total number of seeds in the replicate.

Details

$$cMTG = 1 - t_{50}/t_{max}$$

where t_{50} is the interpolated time at which 50% of the final germination count is reached.

Value

Numeric value in `[0, 1]`. Higher values indicate faster germination.

See Also

`compute_psri_sm`

Examples

```
compute_cmtg(c(20, 23, 25), timepoints = c(3, 5, 7), total_seeds = 25)
compute_cmtg(c(2, 8, 20), timepoints = c(3, 5, 7), total_seeds = 25)
```

`compute_mrg` *Compute Maximum Rate of Germination (MRG)*

Description

Calculates MRG as the scaled average germination rate across consecutive observation intervals.

Usage

```
compute_mrg(germination_counts, timepoints, total_seeds)
```

Arguments

`germination_counts`	Integer vector of cumulative germination counts at each timepoint.
`timepoints`	Numeric vector of observation times (e.g., days). Must be the same length as `germination_counts`.
`total_seeds`	Integer. Total number of seeds in the replicate.

Value

Numeric value >= 0 representing the scaled germination rate. For a single timepoint, returns 0.1 (rate not computable).

See Also

`compute_psri_sm`

Examples

```
compute_mrg(c(5, 15, 20), timepoints = c(3, 5, 7), total_seeds = 25)
compute_mrg(c(10), timepoints = c(3), total_seeds = 25)
```

`compute_msg` *Compute Maximum Stress-adjusted Germination (MSG)*

Description

Calculates MSG as the maximum cumulative germination fraction across all observed timepoints.

Usage

```
compute_msg(germination_counts, total_seeds)
```

Arguments

`germination_counts`	Integer vector of cumulative germination counts at each timepoint.
`total_seeds`	Integer. Total number of seeds in the replicate.

Value

Numeric value in `[0, 1]` representing the maximum germination fraction.

See Also

`compute_psri_sm`

Examples

```
compute_msg(c(5, 15, 20), total_seeds = 25)
compute_msg(c(0, 0, 0), total_seeds = 25)
```

`compute_psri_sm` *Compute Softmax Plant Stress Response Index (PSRI_SM)*

Description

Calculates the softmax-aggregated PSRI from time-series germination data. This method handles zero-value components gracefully through adaptive softmax reweighting, eliminating the zero-collapse failure mode of the geometric mean approach.

Usage

```
compute_psri_sm(
  germination_counts,
  timepoints,
  total_seeds,
  radicle_count = NULL,
  temperature = 0.13,
  return_components = FALSE
)
```

Arguments

`germination_counts`	Integer vector of cumulative germination counts at each timepoint. Must be non-decreasing and not exceed `total_seeds`.
`timepoints`	Numeric vector of observation times (e.g., days). Must be the same length as `germination_counts` and strictly increasing.
`total_seeds`	Integer. Total number of seeds in the replicate.
`radicle_count`	Integer or `NULL`. Number of seeds with radicle emergence at the final observation. If `NULL`, radicle vigor is excluded (3-component PSRI). Default is `NULL`.
`temperature`	Numeric. Softmax temperature parameter. Default is 0.13, calibrated via perplexity targeting. See `calibrate_temperature`.
`return_components`	Logical. If `TRUE`, returns a list containing the PSRI value, all component scores, and softmax weights. Default is `FALSE`.

Details

The softmax PSRI aggregates germination components using weighted summation:

$$PSRI_{SM} = \sum W_i \cdot C_i$$

where $C = [MSG, MRG, cMTG]$ (3-component) or $C = [MSG, MRG, cMTG, RVS]$ (4-component), and weights are:

$$W_i = \frac{\exp(C_i/T)}{\sum_j \exp(C_j/T)}$$

When any component $C_k = 0$, its softmax weight approaches zero and the remaining components absorb its weight, preserving information rather than collapsing the index.

Value

If `return_components = FALSE` (default), a single numeric PSRI value.

If `return_components = TRUE`, a list with:

psri_sm	The softmax PSRI value.
components	Named numeric vector of component scores.
weights	Named numeric vector of softmax weights.
temperature	The temperature parameter used.
n_components	Number of components used (3 or 4).

References

Walne, C.H., Gaudin, A., Henry, W.B., and Reddy, K.R. (2020). In vitro seed germination response of corn hybrids to osmotic stress conditions. *Agrosystems, Geosciences & Environment*, 3(1), e20087. doi:10.1002/agg2.20087

See Also

`compute_msg`, `compute_mrg`, `compute_cmtg`, `compute_rvs`, `softmax_weights`, `calibrate_temperature`

Examples

```
# Basic 3-component PSRI
compute_psri_sm(
  germination_counts = c(5, 15, 20),
  timepoints = c(3, 5, 7),
  total_seeds = 25
)

# With radicle vigor (4-component)
compute_psri_sm(
  germination_counts = c(5, 15, 20),
  timepoints = c(3, 5, 7),
  total_seeds = 25,
  radicle_count = 18
)

# Detailed output
result <- compute_psri_sm(
  germination_counts = c(0, 0, 10),
  timepoints = c(3, 5, 7),
  total_seeds = 25,
  return_components = TRUE
)
result$psri_sm     # > 0 (softmax prevents collapse)
result$components
result$weights
```

`compute_rvs` *Compute Radicle Vigor Score (RVS)*

Description

Calculates a continuous radicle vigor score on a `[0, 1]` scale from radicle emergence counts and total seed counts. Unlike the discrete Radicle Vigor Factor (RVF) in `PSRICalc`, RVS varies continuously.

Usage

```
compute_rvs(radicle_count, total_seeds)
```

Arguments

`radicle_count`	Integer. Number of seeds with visible radicle emergence. If `NULL` or `NA`, returns 0.
`total_seeds`	Integer. Total number of seeds in the replicate.

Value

Numeric value in `[0, 1]`.

See Also

`compute_psri_sm`

Examples

```
compute_rvs(18, total_seeds = 25)
compute_rvs(0, total_seeds = 25)
compute_rvs(NULL, total_seeds = 25)
```

`germination_example` *Example Germination Data for PSRI Softmax Calculation*

Description

A synthetic dataset modeled after prion-exposed germination experiments with corn and barley. Contains time-series germination counts and radicle emergence data across multiple treatments and replicates.

Usage

```
data(germination_example)
```

Format

A data frame with 40 rows and 8 columns:

species Character. Crop species (`"corn"` or `"barley"`).

treatment Character. Treatment group.

replicate Integer. Replicate number (1-4).

total_seeds Integer. Total seeds per replicate (25).

day3 Integer. Cumulative germination count at day 3.

day5 Integer. Cumulative germination count at day 5.

day7 Integer. Cumulative germination count at day 7.

radicle_count Integer. Seeds with radicle emergence at day 7.

Source

Synthetic data modeled after experiments at the Minnesota Center for Prion Research and Outreach (MNPRO), University of Minnesota.

Examples

```
data(germination_example)
head(germination_example)

# Compute PSRI_SM for first row
row1 <- germination_example[1, ]
compute_psri_sm(
  germination_counts = c(row1$day3, row1$day5, row1$day7),
  timepoints = c(3, 5, 7),
  total_seeds = row1$total_seeds,
  radicle_count = row1$radicle_count,
  return_components = TRUE
)
```

`softmax_weights` *Compute Softmax Weights*

Description

Calculates softmax weights for a vector of component scores using a temperature parameter that controls weight concentration.

Usage

```
softmax_weights(scores, temperature = 0.13)
```

Arguments

`scores`	Numeric vector of component scores (at least 2 elements).
`temperature`	Numeric. Temperature parameter. Lower = sharper weights. Default is 0.13.

Details

$$W_i = \frac{\exp(C_i/T)}{\sum_j \exp(C_j/T)}$$

Uses the log-sum-exp trick for numerical stability.

Value

Numeric vector of weights summing to 1. Named if input is named.

See Also

`calibrate_temperature`, `compute_psri_sm`

Examples

```
softmax_weights(c(MSG = 0.8, MRG = 1.0, cMTG = 0.6))
softmax_weights(c(MSG = 0.8, MRG = 0.0, cMTG = 0.7))
softmax_weights(c(0.8, 0.5, 0.3), temperature = 1.0)
```

Index