

# Package: rshift (via r-universe)

September 11, 2024

**Type** Package

**Title** Paleoecology Functions for Regime Shift Analysis

**Version** 3.1.1

**Description** Contains a variety of functions, based around regime shift analysis of paleoecological data. Citations: Rodionov() from Rodionov (2004) <[doi:10.1029/2004GL019448](https://doi.org/10.1029/2004GL019448)> Lanzante() from Lanzante (1996) <[doi:10.1002/\(SICI\)1097-0088\(199611\)16:11%3C1197::AID-JOC89%3E3.0.CO;2-L](https://doi.org/10.1002/(SICI)1097-0088(199611)16:11%3C1197::AID-JOC89%3E3.0.CO;2-L)> Hellinger\_trans from Numerical Ecology, Legendre & Legendre (ISBN 9780444538680) rolling\_autoc from Liu, Gao & Wang (2018) <[doi:10.1016/j.scitotenv.2018.06.276](https://doi.org/10.1016/j.scitotenv.2018.06.276)> Sample data sets lake\_data & lake\_RSI processed from Bush, Silman & Urrego (2004) <[doi:10.1126/science.1090795](https://doi.org/10.1126/science.1090795)> Sample data set January\_PDO from NOAA: <<https://www.ncei.noaa.gov/access/monitoring/pdo/>>.

**Suggests** R.rsp

**VignetteBuilder** R.rsp

**Depends** R (>= 3.5.0)

**Imports** grid, tibble, dplyr, ggplot2

**License** MIT + file LICENSE

**NeedsCompilation** yes

**SystemRequirements** rustc & cargo if building from source

**URL** <https://github.com/alexhroom/rshift>

**BugReports** <https://github.com/alexhroom/rshift/issues>

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.3.1

**Config/rextendr/version** 0.3.1

**Repository** <https://alexhroom.r-universe.dev>

**RemoteUrl** <https://github.com/alexhroom/rshift>

**RemoteRef** HEAD

**RemoteSha** a7b5cb162138e78e5e2f8fdab6669b90ccb98e35

## Contents

absolute_to_percentage . . . . .	2
Hellinger_trans . . . . .	3
January_PDO . . . . .	3
lake_data . . . . .	4
lake_RSI . . . . .	4
Lanzante . . . . .	5
regime_means . . . . .	5
Rodionov . . . . .	6
rolling_autoc . . . . .	7
RSI_graph . . . . .	7
rust_regime_means . . . . .	8
rust_rodionov . . . . .	8

**Index** **9**

---

absolute\_to\_percentage

*Converts absolute abundance data to a percentage of total abundance for each site*

---

### Description

Converts absolute abundance data to a percentage of total abundance for each site

### Usage

```
absolute_to_percentage(data, col, site)
```

### Arguments

data	The dataframe to be used.
col	The column that change is being measured on.
site	The column containing the site of each sample.

### Value

The 'data' dataframe with an added 'percentage' column.

---

Hellinger_trans	<i>Hellinger transform</i>
-----------------	----------------------------

---

**Description**

Hellinger transforms data (Legendre and Legendre, Numerical Ecology)

**Usage**

```
Hellinger_trans(data, col, site)
```

**Arguments**

data	The dataframe to be used.
col	The column that change is being measured on.
site	The column containing the site of each sample.

**Value**

The 'data' dataframe with an added 'hellinger\_trans\_vals' column.

---

January_PDO	<i>Pacific Decadal Oscillation in January</i>
-------------	---

---

**Description**

A dataset containing January PDO values. Subset of the data from NOAA: <<https://www.ncei.noaa.gov/access/monitoring/pd>

**Usage**

```
data(January_PDO)
```

**Format**

A data frame with 104 rows and 2 variables

**Details**

- PDO - Pacific Decadal Oscillation in January for the given year.
- Age - the year for which the PDO was measured.

---

lake\_data

*DCA-ordinated pollen data from Lake Consuelo*

---

**Description**

A dataset containing pre-processed DCA-ordinated data from Bush, Silman & Urrego (2004) <doi:10.1126/science.1090795>

**Usage**

```
data(lake_data)
```

**Format**

A data frame with 39 rows and 2 variables

**Details**

- DCA1 - DCA values for each timepoint from the raw dataset.
- Age - timepoint of each sample that has been DCA-ordinated.

---

lake\_RSI

*DCA-ordinated pollen data from Lake Consuelo with RSI values*

---

**Description**

A dataset containing pre-processed DCA-ordinated data from Bush, Silman & Urrego (2004) <doi:10.1126/science.1090795>  
This data has been processed using `Rodionov(lake_data, "DCA1", "Age", l=5, merge=TRUE)`

**Usage**

```
data(lake_RSI)
```

**Format**

A data frame with 39 rows and 3 variables

**Details**

- DCA1 - DCA values for each timepoint from the raw dataset.
- Age - timepoint of each sample that has been DCA-ordinated.
- RSI - Regime Shift Index (see docs for `Rodionov()`) for each timepoint.

---

Lanzante	<i>Lanzante L-test</i>
----------	------------------------

---

**Description**

performs the L-method for detection of regime shifts (Lanzante, 1996)

**Usage**

```
Lanzante(data, col, time, p = 0.05, merge = FALSE)
```

**Arguments**

data	The dataframe to be used.
col	The column we are measuring change on.
time	The column containing time units (e.g. age of a subsample)
p	The largest p-value you want to check regime shifts for. Defaults to $p = 0.05$ .
merge	Sets the result to be either a regime-shift only table (if FALSE), or an addition to the original table (if TRUE)

**Value**

If `merge = FALSE` (default), produces a 2-column table of time (the time value for each regime shift) and `p` (the p-value for each regime shift). If `merge = TRUE`, returns the original dataset with an extra p-value column, giving the p-value for each time unit - 0 for non-shift years.

**Examples**

```
Lanzante(lake_data, "DCA1", "Age")
Lanzante(lake_data, "DCA1", "Age", p=0.10, merge=TRUE)
```

---

regime_means	<i>Calculate means for each regime</i>
--------------	--

---

**Description**

calculates the mean for each regime in a regime shift analysis.

**Usage**

```
regime_means(data, col, rsi)
```

**Arguments**

data	The dataframe that will be used.
col	The column we are measuring change on.
rsi	The column containing RSI values.

**Value**

A vector of the mean value for each regime.

**Examples**

```
regime_means(lake_RSI, "DCA1", "RSI")
```

---

 Rodionov

*Rodionov (2004)'s STARS algorithm*


---

**Description**

performs STARS analysis (Rodionov, 2004) on a dataset

**Usage**

```
Rodionov(data, col, time, l, prob = 0.05, startrow = 1, merge = FALSE)
```

**Arguments**

data	The dataframe to be used.
col	The column we are measuring change on.
time	The column containing time units (e.g. age of a subsample)
l	The cut-off length of a regime; affects sensitivity (see Rodionov, 2004)
prob	The p-value for significance of a regime shift. Defaults to $p = 0.05$ .
startrow	What row the analysis starts at. Defaults to 1.
merge	Sets the result to be either a regime-shift only table (if FALSE), or an addition to the original table (if TRUE)

**Value**

If `merge = FALSE` (default), produces a 2-column table of time (the time value for each regime shift) and RSI (the regime shift index for each regime shift). If `merge = TRUE`, returns the original dataset with an extra RSI column, giving the regime shift index for each time unit - 0 for non-shift years.

**Examples**

```
Rodionov(lake_data, "DCA1", "Age", l=5)
Rodionov(lake_data, "DCA1", "Age", l=5, prob=0.01, startrow=2, merge=TRUE)
```

---

rolling_autoc	<i>Rolling autocorrelation</i>
---------------	--------------------------------

---

**Description**

finds lag-1 autocorrelation in a rolling window; can be used to predict resilience (Liu, Gao, & Wang, 2018)

**Usage**

```
rolling_autoc(data, col, l)
```

**Arguments**

data	The dataframe that will be used.
col	The column we are measuring change on.
l	The time interval (no. of columns) used in the autocorrelation.

**Value**

A table of rolling lag-1 autocorrelation values.

---

RSI_graph	<i>Regime Shift Index graph</i>
-----------	---------------------------------

---

**Description**

creates two graphs, one of data and one of the RSI, as seen in Rodionov (2004)

**Usage**

```
RSI_graph(data, col, time, rsi, mean_lines = FALSE)
```

**Arguments**

data	The dataframe that will be used.
col	The column we are measuring change on.
time	The column containing time units (e.g. age of a subsample)
rsi	The column containing RSI values (for best visualisation use Rodionov() with merge=TRUE)
mean_lines	If true, add lines over the data indicating the mean of each regime.

**Value**

Two graphs, one on top of the other; one of col against time and one of RSI against time.

**Examples**

```
RSI_graph(lake_RSI, "DCA1", "Age", "RSI")
```

---

rust_regime_means	<i>Calculates the mean for each regime in a regime shift analysis.</i>
-------------------	--

---

**Description**

Calculates the mean for each regime in a regime shift analysis.

**Usage**

```
rust_regime_means(col, rsi)
```

**Arguments**

col	The column we are measuring change on.
rsi	The column containing RSI values.

---

rust_rodionov	<i>Calculate STARS RSI points and return to R as a vector</i>
---------------	---

---

**Description**

Calculate STARS RSI points and return to R as a vector

**Usage**

```
rust_rodionov(vals, t_crit, l)
```

**Arguments**

vals	The column we are measuring change on
t_crit	The critical value of a t-distribution at the desired p-value
l	The cut-off length of a regime; affects sensitivity



# Index

## \* datasets

January\_PDO, 3

lake\_data, 4

lake\_RSI, 4

absolute\_to\_percentage, 2

Hellinger\_trans, 3

January\_PDO, 3

lake\_data, 4

lake\_RSI, 4

Lanzante, 5

regime\_means, 5

Rodionov, 6

rolling\_autoc, 7

RSI\_graph, 7

rust\_regime\_means, 8

rust\_rodionov, 8