

# Package: lakthesis (via r-universe)

August 24, 2024

**Title** Consensus Seriation for Binary Data

**Version** 0.0.2

**Description** Determining consensus seriations for binary incidence matrices, using a two-step process of Procrustes-fit correspondence analysis for heuristic selection of partial seriations and iterative regression to establish a single consensus. Contains the Lakthesis Calculator, a graphical platform for identifying seriated sequences. Collins-Elliott (2024) <<https://volweb.utk.edu/~scoll146/sceLakthesis.pdf>>.

**License** GPL (>= 3)

**Imports** stats, readr, ca, ggplot2, Rdpack, shiny, shinydashboard, bslib

**RdMacros** Rdpack

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.1

**Depends** R (>= 2.10)

**LazyData** true

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

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

**RemoteUrl** <https://github.com/scollinselliott/lakthesis>

**RemoteRef** HEAD

**RemoteSha** 2fca746ac975a34b55249ce8c27ba8e5c567d5eb

## Contents

conc_col . . . . .	2
conc_kappa . . . . .	3
element_eval . . . . .	3

im_long . . . . .	4
im_merge . . . . .	5
im_read_csv . . . . .	6
im_ref . . . . .	7
lakhesize . . . . .	7
LC . . . . .	9
plot.procrustean . . . . .	10
plot.strand . . . . .	11
qfStrands . . . . .	12
quattrofontanili . . . . .	12
spearman_sq . . . . .	13
strand_add . . . . .	14
strand_extract . . . . .	14
strand_suppress . . . . .	15
<b>Index</b>	<b>17</b>

---

conc_col	<i>Kendall-Doran Concentration</i>
----------	------------------------------------

---

## Description

The Kendall-Doran measure of concentration (Kendall 1963; Doran 1971). In a seriated matrix, this function computes the total number cells between the first and last non-zero value, column by column.

## Usage

```
conc_col(obj)
```

## Arguments

obj            A seriated binary matrix.

## Value

The measure of concentration.

## References

Doran J (1971). “Computer Analysis of Data from the la Tène Cemetery at Münsingen-Rain.” In Hodson FR, Kendall DG, Táutu P (eds.), *Mathematics in the Archaeological and Historical Sciences*, 422–431. Edinburgh University Press, Edinburgh.

Kendall DG (1963). “A Statistical Approach to Flinders Petrie’s Sequence Dating.” *Bulletin of the International Statistical Institute*, **40**, 657–680.

**Examples**

```
data("quattrofontanili")
conc_col(quattrofontanili)
```

---

conc_kappa	<i>Kappa Concentration</i>
------------	----------------------------

---

**Description**

The concentration coefficient  $\kappa$ , which extends the Kendall-Doran measure of concentration to include rows and then weights the total measure by the total sum of values in the matrix. See [concentration.col](#).

**Usage**

```
conc_kappa(obj)
```

**Arguments**

obj            A seriated binary matrix.

**Value**

The  $\kappa$  coefficient of concentration.

**Examples**

```
data("quattrofontanili")
conc_kappa(quattrofontanili)
```

---

element_eval	<i>Evaluating Element Fit</i>
--------------	-------------------------------

---

**Description**

Performs a goodness-of-fit test on individual row and column elements using deviance, using a quadratic-logistic model to fit row and column occurrences. In the case of perfect separation of 0/1 values, an NA value is assigned. Results are reported as  $p$  values for each row and column.

**Usage**

```
element_eval(obj)
```

```
## S3 method for class 'incidence_matrix'
element_eval(obj)
```

**Arguments**

obj                    A seriated binary matrix.

**Value**

A list containing results in data frames for row and column elements:

- RowFit a data frame containing
  - id Row element
  - p.val  $p$  values of the row elements
- ColFit a data frame containing
  - id Column element
  - p.val  $p$  values of the column elements

**Examples**

```
data("quattrofontanili")
element_eval(quattrofontanili)
```

---

 im\_long

---

*Convert Incidence Matrix to Pairs (Long Format)*


---

**Description**

Take an incidence matrix and convert it to a data frame of two columns, where the first column represents the row elements of the incidence matrix and the second column represents the column elements of the incidence matrix. Each row pair represents the incidence (or occurrence) of that row and column element together.

**Usage**

```
im_long(obj)

## S3 method for class 'matrix'
im_long(obj)

## S3 method for class 'incidence_matrix'
im_long(obj)
```

**Arguments**

obj                    An incidence matrix.

**Value**

A data frame of two columns (row and column of the incidence matrix), in which row of the data frame represents a pair of an

**Examples**

```
data(quattrofontanili)
qf <- im_long(quattrofontanili)

# to export for uploading into the Lakthesis Calculator, use write.table() to
# remove both row and column names:

# write.table(qf, file = 'qf.csv', row.names = FALSE, col.names = FALSE, sep = ",")
```

---

im_merge	<i>Merge Two Incidence Matrices</i>
----------	-------------------------------------

---

**Description**

From two incidence matrices, create a single incidence matrix. Matrices may contain same row or column elements.

**Usage**

```
im_merge(obj1, obj2)

## S3 method for class 'matrix'
im_merge(obj1, obj2)

## S3 method for class 'incidence_matrix'
im_merge(obj1, obj2)
```

**Arguments**

obj1, obj2      Two incidence matrices of any size.

**Value**

A single incidence matrix.

**Examples**

```
data(quattrofontanili)
qf1 <- quattrofontanili[1:20, 1:40]
qf1 <- qf1[rowSums(qf1) != 0, colSums(qf1) != 0]

qf2 <- quattrofontanili[30:50, 20:60]
qf2 <- qf2[rowSums(qf2) != 0, colSums(qf2) != 0]

im_merge(qf1, qf2)
```

---

`im_read_csv`*Read csv File to Incidence Matrix*

---

## Description

Wrapper around the `read_csv` function from the `readr` package (Wickham et al. 2024). Read a `.csv` file in which the first column represents row elements and the second column represents column elements, and convert it into an incidence matrix.

## Usage

```
im_read_csv(  
  filename,  
  header = FALSE,  
  characterencoding = "iso-8859-1",  
  remove.hapax = FALSE  
)  
  
## S3 method for class 'incidence_matrix'  
plot(im_seriated)
```

## Arguments

<code>filename</code>	The filename to uploaded (must be in <code>.csv</code> format).
<code>header</code>	If the <code>.csv</code> file contains a header. Default is <code>FALSE</code> .
<code>characterencoding</code>	File encoding as used by <code>locale</code> . Default is <code>"iso-8859-1"</code> to handle special characters.
<code>remove.hapax</code>	Remove any row or column which has a sum of 1 (i.e., is only attested once), since they do not directly contribute to the result of the seriation. Default is <code>FALSE</code> .

## Value

A matrix of binary values (0 = row/column occurrence is absence; 1 = row/column occurrence is present).

## References

Wickham H, Hester J, Bryan J (2024). *readr: Read Rectangular Text Data*. R package version 2.1.5, <https://github.com/tidyverse/readr>, <https://readr.tidyverse.org>.

---

im_ref	<i>Create Reference Matrix</i>
--------	--------------------------------

---

**Description**

Create an ideal reference matrix of well-seriated values of the same size as the input matrix.

**Usage**

```
im_ref(obj)

## S3 method for class 'matrix'
im_ref(obj)
```

**Arguments**

obj                    A matrix of size  $n \times k$ .

**Value**

A matrix of size  $n \times k$  with 1s along the diagonal. If  $n > k$ , 1s are placed from cell  $(i, i)$  to  $(i, i + k - n)$ , with 0 in all other cells.

**Examples**

```
im_ref(matrix(NA, 5, 5))
im_ref(matrix(1, 7, 12))
```

---

lakhesize	<i>Lakhesize</i>
-----------	------------------

---

**Description**

This function returns the row and column consensus seriation for a list object of the strands class, containing their rankings and coefficients of association and concentration.

**Usage**

```
lakhesize(strands, ...)
```

```
## S3 method for class 'strands'
lakhesize(strands, pbar = TRUE)
```

```
## Default S3 method:
lakhesize(strands, pbar = TRUE)
```

```
## S3 method for class 'lakthesis'
plot(result, display = "im_seriated")
```

### Arguments

`strands` A list of the strands class (see [add\\_strand](#)).

`pbar` Displaying a progress bar. Default is TRUE.

### Details

Consensus seriation is achieved by iterative simple linear regression to handle NA vales in each strand. To initialize, a regression is performed pairwise, with every strand as the dependent  $y$  variate and every other strand as the independent  $x$  variate. The independent variate's rankings are then regressed onto  $f(x) = \hat{\beta}_1 x + \hat{\beta}_0$ . If  $y \neq f(x)$ , the mean of  $y$  and  $f(x)$  is used. Then, the values of dependent variate and those of regressed independent are re-ranked, which serves as the dependent variate on the next iteration. The process is repeated, regressing each strand which yields the lowest concentration measure.

### Value

A list of class `lakthesis` containing the following:

- `row` A seriated vector of row elements.
- `col` A seriated vector of column elements
- `coef` A data frame containing the following columns:
  - `Strand` The number of the strand.
  - `Agreement` The measure of agreement, i.e., how well each strand accords with the consensus seriation. Using the square of Spearman's rank correlation coefficient,  $\rho^2$ , between each strand and the consensus ranking, agreement is computed as the product of  $\rho^2$  for their row and column rankings,  $\rho_r^2 \rho_c^2$ .
  - `Concentration` the concentration coefficient  $\kappa$ , which provides a measure of the optimality of each strand (see [kappa.coef](#)).
- `im_seriated` The seriated incidence matrix, of class `incidence_matrix`.

### Examples

```
data("qfStrands")
x <- lakhesize(qfStrands, pbar = FALSE)
# summary(x)
```



**Description**

Launch Lakthesis Calculator, a graphical interface to explore binary matrices via correspondence analysis, select potentially well-seriated sequences, and perform consensus seriation. Interface is made with `ggplot2`, `shiny`, `shinydashboard`, and `bslib` (Wickham 2016; Chang et al. 2024; Chang and Borges Ribeiro 2021; Sievert et al. 2024).

**Usage**

```
LC()
```

**Details**

Input is done in the calculator, via a "long" format a two-column .csv file giving pairs of row and column incidences. See `im_read_csv` for details. Conversion of a pre-existing incidence matrix to long format can be performed with `im_long`.

Results can be downloaded from the calculator as an .rds file containing a list of the following:

- consensus The consensus seriations, PCA, coefficients of agreement and concentration, and seriated incidence matrix. (`Lakthesize`).
- strands The strands selected by the investigator.

**Value**

Opens the Lakthesis Calculator.

**References**

Chang W, Borges Ribeiro B (2021). *shinydashboard: Create Dashboards with 'Shiny'*. <https://CRAN.R-project.org/package=shinydashboard>.

Chang W, Cheng J, Allaire JJ, Sievert C, Schloerke B, Xie Y, Allen J, McPherson J, Dipert A, Borges B (2024). *shiny: Web Application Framework for R*. R package version 1.8.1.9001; <https://github.com/rstudio/shiny>, <https://shiny.posit.co>.

Sievert C, Cheng J, Aden-Buie G (2024). *bslib: Custom 'Bootstrap' 'Sass' Themes for 'shiny' and 'rmarkdown'*. R package version 0.7.0, <https://github.com/rstudio/bslib>, <https://rstudio.github.io/bslib/>.

Wickham H (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer, New York.

---

plot.procrustean      *Correspondence Analysis with Procrustes Fitting*

---

### Description

Fit scores of correspondence analysis on an incidence matrix to those produced by reference matrix which contain an ideal seriation using a Procrustes method (on the reference matrix, see [im\\_ref](#)). Rotation is determined by minimizing Euclidean distance from each row score to the nearest reference row score. Correspondence analysis is performed using the [ca](#) package (Nenadic and Greenacre 2007).

### Usage

```
## S3 method for class 'procrustean'  
plot(result)  
  
ca_procrustes(obj)  
  
## S3 method for class 'matrix'  
ca_procrustes(obj)  
  
## S3 method for class 'incidence_matrix'  
ca_procrustes(obj)
```

### Arguments

obj                    An incidence matrix of size  $n \times k$ .

### Value

A list object of class `strand` containing the following:

- `ref` The Procrustes-fit coordinates of the scores of the reference seriation.
- `x` The Procrustes-fit coordinates of the row scores of the data.
- `y` The Procrustes-fit coordinates of the column scores of the data.

### References

Nenadic O, Greenacre MJ (2007). “Correspondence Analysis in R, with Two- and Three-dimensional Graphics: The `ca` Package.” *Journal of Statistical Software*, **20**, 1–13. doi:10.18637/jss.v020.i03.

### Examples

```
data("quattrofontanili")  
s <- ca_procrustes(quattrofontanili)  
# print(s)
```

---

plot.strand

*Seriate Using Reference Curve for Procrustes-Fit CA Scores*


---

### Description

Obtain a ranking of row and column scores projected onto a reference curve of an ideal seriation (row and column scores are ranked separately). Scores of correspondence analysis have been fit to those produced by reference matrix contain an ideal seriation using a Procrustes method, projecting them. Rotation is determined by minimizing Euclidean distance from each row score to the nearest reference row score. Correspondence analysis is performed using the `ca` package (Nenadic and Greenacre 2007).

### Usage

```
## S3 method for class 'strand'
plot(strand, display = "ca")

ca_procrustes_ser(obj, samples = 10^5)

## S3 method for class 'incidence_matrix'
ca_procrustes_ser(obj, samples = 10^5)

## S3 method for class 'matrix'
ca_procrustes_ser(obj, samples = 10^5)
```

### Arguments

<code>obj</code>	An incidence matrix of size $n \times k$ .
<code>samples</code>	Number of samples to use for plotting points along polynomial curve. Default is $10^5$ .

### Value

A list of class `strand` containing the following:

- `$dat` A data frame with the following columns:
  - `Procrustes1`, `Procrustes2` The location of the point on the biplot after fitting.
  - `CurveIndex` The orthogonal projection of the point onto the reference curve, given as the index of the point sampled along  $y = \beta_2 x^2 + \beta_0$ .
  - `Distance` The squared Euclidean distance of the point to the nearest point on the reference curve.
  - `Rank` The ranking of the row or column, a range of `1:nrow` `` and `1:ncol` ``.
  - `Type` Either `row` or `col`.
  - `sel` Data frame column used in shiny app to indicate whether point is selected in biplot/curve projection.
- `$im_seriated` The seriated incidence matrix, of class `incidence_matrix`.

## References

Nenadic O, Greenacre MJ (2007). “Correspondence Analysis in R, with Two- and Three-dimensional Graphics: The ca Package.” *Journal of Statistical Software*, **20**, 1–13. doi:10.18637/jss.v020.i03.

## Examples

```
data("quattrofontanili")
s <- ca_procrustes_ser(quattrofontanili)
# print(s)
# summary(s)
```

---

qfStrands

*Quattro Fontanili - Strands*

---

## Description

Three seriated strands selected from [quattrofontanili](#) data, identified by the package author as an example for the documentation of functions.

## Usage

```
data("qfStrands")
```

## Format

A list containing data frames output by [ca.procrustes.curve](#).

## Examples

```
data("qfStrands")
print(qfStrands)
```

---

quattrofontanili

*Quattro Fontanili*

---

## Description

The seriation of tombs from necropoleis at Veii, primarily Quattro Fontanili, but also Valle la Fata, Vaccareccia, and Picazzano, in southern Etruria, established by Close-Brooks and Ridgway (1979).

## Usage

```
data("quattrofontanili")
```

**Format**

A seriated incidence matrix of 81 rows (tombs) and 82 columns (types).

Data entered from Close-Brooks and Ridgway (1979), an English translation of the authors' original publication in *Notizie degli Scavi* (1963). Descriptions of types may be found in that paper.

**References**

Close-Brooks J, Ridgway D (1979). "Veii in the Iron Age." In Ridgway D, Ridgway FR (eds.), *Italy Before the Romans*, 95–127. Academic Press, London.

**Examples**

```
data("quattrofontanili")
print(quattrofontanili)
```

---

spearman\_sq

*Spearman Correlation Squared*

---

**Description**

The square of Spearman's rank correlation coefficient applied to two rankings (Spearman 1904). Rows with NA values are automatically removed.

**Usage**

```
spearman_sq(r1, r2)
```

**Arguments**

r1, r2            Two vectors of paired ranks.

**Value**

The square of Spearman's rank correlation coefficient with NA values removed.

**References**

Spearman C (1904). "The Proof and Measurement of Association between Two Things." *American Journal of Psychology*, **15**, 72–101. doi:10.2307/1412159.

**Examples**

```
# e.g., for two partial seriations:
x <- c(1, 2, 3, 4, NA, 5, 6, NA, 7.5, 7.5, 9)
y <- c(23, 17, 19, NA, 21, 22, 25, 26, 27, 36, 32)
spearman_sq(x, y)
```

---

strand_add	<i>Add Strand to List of Strands</i>
------------	--------------------------------------

---

**Description**

Given a list of strands, remove a row or column element and re-run seriation by correspondence analysis with Procrustes fitting ([ca\\_procrustes\\_ser](#)) to generate a new list of strands that exclude the specified elements. If the resulting strand lacks sufficient points to perform correspondence analysis, that strand is deleted in the output.

**Usage**

```
strand_add(strand, ...)
```

```
## S3 method for class 'strand'
```

```
strand_add(strand, strands)
```

**Arguments**

strand	An object of class strand returned by <a href="#">ca_procrustes_ser</a> .
strands	A list of strands.

**Value**

A list of class strands.

---

strand_extract	<i>Strand Extract</i>
----------------	-----------------------

---

**Description**

From a list of strands produced by [ca\\_procrustes\\_ser](#), extract two matrices containing the ranks of the rows and columns. The row/column elements are contained in the rows, and the strands are contained in the columns. NA values are entered where a given row/column element is missing from that strand.

**Usage**

```
strand_extract(strands, ...)
```

```
## S3 method for class 'strands'
```

```
strand_extract(strands)
```

**Arguments**

strands	A list of class strands.
---------	--------------------------

**Value**

A list of two matrices:

- Row A matrix of the ranks of the row elements.
- Col A matrix of the ranks of the column elements.

**Examples**

```
data("quattrofontanili")
data("qfStrands")
strand_extract(qfStrands)
```

---

strand_suppress	<i>Suppress Element from Strands</i>
-----------------	--------------------------------------

---

**Description**

Given a list of strands, remove a row or column element and re-run seriation by correspondence analysis with Procrustes fitting ([ca\\_procrustes\\_ser](#)) to generate a new list of strands that exclude the specified elements. If the resulting strand lacks sufficient points to perform correspondence analysis, that strand is deleted in the output.

**Usage**

```
strand_suppress(strands, ...)
```

```
## S3 method for class 'strands'
strand_suppress(strands, ...)
```

```
## Default S3 method:
strand_suppress(strands, elements)
```

**Arguments**

strands	A list of class strands.
elements	A vector of one or more row or column ids to suppress.

**Value**

A list of the strands.

**Examples**

```
data("quattrofontanili")
data("qfStrands")
strand_suppress(qfStrands, "QF II 15-16")

strand_suppress(qfStrands, c("QF II 15-16", "I", "XIV"))
```



# Index

## \* datasets

qfStrands, [12](#)  
quattrofontanili, [12](#)

add\_strand, [8](#)

ca, [10](#), [11](#)

ca.procrustes.curve, [12](#)

ca.procrustes(plot.procrustean), [10](#)

ca.procrustes\_ser, [14](#), [15](#)

ca.procrustes\_ser(plot.strand), [11](#)

conc\_col, [2](#)

conc\_kappa, [3](#)

concentration.col, [3](#)

element\_eval, [3](#)

im\_long, [4](#), [9](#)

im\_merge, [5](#)

im\_read\_csv, [6](#), [9](#)

im\_ref, [7](#), [10](#)

kappa.coef, [8](#)

lakhesize, [7](#), [9](#)

LC, [9](#)

locale, [6](#)

plot.incidence\_matrix(im\_read\_csv), [6](#)

plot.lakthesis(lakhesize), [7](#)

plot.procrustean, [10](#)

plot.strand, [11](#)

qfStrands, [12](#)

quattrofontanili, [12](#), [12](#)

read\_csv, [6](#)

spearman\_sq, [13](#)

strand\_add, [14](#)

strand\_extract, [14](#)

strand\_suppress, [15](#)