

# Package ‘coRanking’

March 20, 2023

**Title** Co-Ranking Matrix

**Version** 0.2.4

**Description** Calculates the co-ranking matrix to assess the quality of a dimensionality reduction.

**URL** <https://www.guido-kraemer.com/software/coranking/>

**BugReports** <https://github.com/gdkrmr/coRanking/issues>

**License** GPL-3 | file LICENSE

**Encoding** UTF-8

**Imports** methods, graphics, stats

**RoxygenNote** 7.2.3

**Collate** 'coranking-package.R' 'coranking.R' 'coranking\_internals.R'  
'lcmc.R' 'criteria.R' 'image.R' 'rankmatrix.R' 'submatrix.R'  
'r\_nx.R'

**Suggests** testthat, knitr, rmarkdown, Rtsne, scatterplot3d

**VignetteBuilder** knitr

**NeedsCompilation** yes

**Author** Guido Kraemer [aut, cre] (<<https://orcid.org/0000-0003-4865-5041>>)

**Maintainer** Guido Kraemer <[guido.kraemer@uni-leipzig.de](mailto:guido.kraemer@uni-leipzig.de)>

**Repository** CRAN

**Date/Publication** 2023-03-20 21:20:02 UTC

## R topics documented:

coRanking-package . . . . .	2
AUC_In_K . . . . .	3
coranking . . . . .	4
imageplot . . . . .	5
LCMC . . . . .	6
plot_R_NX . . . . .	6
Q_NX . . . . .	7
rankmatrix . . . . .	8
R_NX . . . . .	9

---

coRanking-package      *Methods for the co-ranking matrix*

---

## Description

coRanking provides methods for the calculation of the co-ranking matrix and derived measures to assess the quality of a dimensionality reduction

## Details

This package provides functions for calculating the co-ranking matrix, plotting functions and some derived measures for quality assessment of dimensionality reductions.

Funding provided by the Department for Biogeochemical Integration, Empirical Inference of the Earth System Group, at the Max Plack Institute for Biogeochemistry, Jena.

## Author(s)

**Maintainer:** Guido Kraemer <guido.kraemer@uni-leipzig.de> ([ORCID](#))

## References

Chen, L., Buja, A., 2006. Local Multidimensional Scaling for Nonlinear Dimension Reduction, Graph Layout and Proximity Analysis.

Lee, J.A., Lee, J.A., Verleysen, M., 2009. Quality assessment of dimensionality reduction: Rank-based criteria. *Neurocomputing* 72.

Lueks, W., Mokbel, B., Biehl, M., & Hammer, B. (2011). How to Evaluate Dimensionality Reduction? - Improving the Co-ranking Matrix. *ArXiv:1110.3917 [Cs]*. <http://arxiv.org/abs/1110.3917>

Lee, J. A., Peluffo-Ordóñez, D. H., & Verleysen, M., 2015. Multi-scale similarities in stochastic neighbour embedding: Reducing dimensionality while preserving both local and global structure. *Neurocomputing*, 169, 246–261. <https://doi.org/10.1016/j.neucom.2014.12.095>

## See Also

Useful links:

- <https://www.guido-kraemer.com/software/coranking/>
- Report bugs at <https://github.com/gdkrmr/coRanking/issues>

AUC<sub>ln</sub>K*Area under the R<sub>NX</sub> curve***Description**

Area under the  $R_{NX}(K)$  curve when  $K$  is put on a logarithmic scale.

**Usage**

AUC<sub>ln</sub>K(R<sub>NX</sub>)

**Arguments**

R<sub>NX</sub>                      The R<sub>NX</sub> curve, a vector of values

**Details**

It is calculated as:

$$AUC_{\ln K}(R_{NX}(K)) = \left( \sum_{K=1}^{N-2} R_{NX}(K)/K \right) / \left( \sum_{K=1}^{N-2} 1/K \right)$$

**Value**

A value, the area under the curve.

**Author(s)**

Guido Kraemer

**References**

Lee, J. A., Peluffo-Ordóñez, D. H., & Verleysen, M., 2015. Multi-scale similarities in stochastic neighbour embedding: Reducing dimensionality while preserving both local and global structure. *Neurocomputing*, 169, 246–261. <https://doi.org/10.1016/j.neucom.2014.12.095>

---

`coranking`*Co-Ranking Matrix*

---

**Description**

Calculate the co-ranking matrix to assess the quality of a dimensionality reduction.

**Usage**

```
coranking(  
  Xi,  
  X,  
  input_Xi = c("data", "dist", "rank"),  
  input_X = input_Xi,  
  use = "C"  
)
```

**Arguments**

<code>Xi</code>	high dimensional data
<code>X</code>	low dimensional data
<code>input_Xi</code>	type of input of Xi (see. details)
<code>input_X</code>	type of input of X (see. details)
<code>use</code>	R or C backend

**Details**

Calculate the coranking matrix, to assess the quality of a dimensionality reduction. `Xi` is input in high dimensions, `X` is input in low dimensions the type of input is given in `input_Xi` and `input_X`, they can be one of `c('data', 'dist', 'rank')`.

**Value**

a matrix of class 'coranking'

**Author(s)**

Guido Kraemer

**See Also**

[rankmatrix](#)

---

`imageplot`*Image function for the co-ranking matrix*

---

**Description**

Plots the co-ranking matrix nicely

**Usage**

```
imageplot(  
  Q,  
  lwd = 2,  
  bty = "n",  
  main = "co-ranking matrix",  
  xlab = expression(R),  
  ylab = expression(Ro),  
  col = colorRampPalette(colors = c("gray85", "red", "yellow", "green", "blue"))(100),  
  axes = FALSE,  
  legend = TRUE,  
  ...  
)
```

**Arguments**

<code>Q</code>	of class <code>coranking</code> .
<code>lwd</code>	linewidth in legend
<code>bty</code>	boxtype of legend
<code>main</code>	title of plot
<code>xlab</code>	label of the x axis
<code>ylab</code>	label of the y axis
<code>col</code>	a palette for coloring
<code>axes</code>	logical draw axes
<code>legend</code>	if T plot a legend.
<code>...</code>	parameters for the <code>image</code> function.

**Details**

Plots the co-ranking matrix nicely for visual inspection. uses the `image` function internally, `...` is passed down to the `image` function. The values in the co-ranking matrix are logscaled for better contrast.

**Author(s)**

Guido Kramer

---

 LCMC

*The local continuity meta-criterion*


---

**Description**

Calculate the local continuity meta-criterion from a co-ranking matrix.

**Usage**

```
LCMC(Q, K = 1:nrow(Q))
```

**Arguments**

Q                    a co-ranking matrix  
 K                    vector of integers describing neighborhood size

**Details**

The local continuity meta-criterion (Chen and Buja, 2006) is defined as

$$LCMC = \frac{K}{1-N} + \frac{1}{NK} \sum_{(k,l) \in UL_K} q_{kl}$$

Higher values mean a better performance of the dimensionality reduction.

**Value**

A number, the local continuity meta-criterion

**Author(s)**

Guido Kraemer

---

 plot\_R\_NX

*Plot the R\_NX(K) curve*


---

**Description**

Produces a plot with the  $R_{NX}(K)$  curves from the arguments

**Usage**

```
plot_R_NX(R_NXs, pal = grDevices::palette(), ylim = c(0, 0.9), ...)
```

**Arguments**

R_NXs	A list of R_NX curves, names from the list will appear in the legend
pal	a vector of colors
ylim	set the y-axis limits of the plot
...	options for the plotting function

**Value**

Nothing, produces a plot.

**Author(s)**

Guido Kraemer

**References**

Lee, J. A., Peluffo-Ordóñez, D. H., & Verleysen, M., 2015. Multi-scale similarities in stochastic neighbour embedding: Reducing dimensionality while preserving both local and global structure. *Neurocomputing*, 169, 246–261. <https://doi.org/10.1016/j.neucom.2014.12.095>

---

Q\_NX

*The  $Q_{NX}(K)$  criterion*

---

**Description**

A curve indicating the percentage of points that are mild in- and extrusions or keep their rank.

**Usage**

Q\_NX(Q)

**Arguments**

Q                      a co-ranking matrix

**Details**

$$Q_{NX}(K) = \frac{1}{KN} \sum_{k=1}^K \sum_{l=1}^K Q_{kl}$$

**Value**

A vector with the values for Q\_NX(K)

**Author(s)**

Guido Kraemer

**References**

Lueks, W., Mokbel, B., Biehl, M., & Hammer, B. (2011). How to Evaluate Dimensionality Reduction? - Improving the Co-ranking Matrix. ArXiv:1110.3917 [Cs]. <http://arxiv.org/abs/1110.3917>

---

`rankmatrix`*Rank matrix*

---

**Description**

Replaces the elements of  $X$  with their rank in the column vector of the distance matrix

**Usage**

```
rankmatrix(X, input = c("data", "dist"), use = "C")
```

**Arguments**

<code>X</code>	data, dist object, or distance matrix
<code>input</code>	type of input
<code>use</code>	if 'C' uses the compiled library, else uses the native R code

**Details**

Each column vector in the distance matrix (or the distance matrix computed from the input) is replaced by a vector indicating the rank of the distance inside that vector.

This is a computation step necessary for the co-ranking matrix and provided mainly so that the user has the possibility to save computation time.

**Value**

returns a matrix of class 'rankmatrix'

**Author(s)**

Guido Kraemer



R\_NX

*The R\_NX(K) criterion***Description**

A curve indicating the improvement of the embedding over a random embedding for the neighborhood size  $K$ . Values range from 0, for a random embedding, to 1 for a perfect embedding.

**Usage**

R\_NX(Q)

**Arguments**

Q a co-ranking matrix

**Details**

$R_{NX}(K)$  is calculated as follows:

$$Q_{NX}(K) = \sum_{1 \leq k \leq K} \sum_{1 \leq l \leq K} \frac{q_{kl}}{KN}$$

Counts the upper left  $K$ -by- $K$  block of  $Q$ , i.e. it considers the preserved ranks on the diagonal and the permutations within a neighborhood.

$$R_{NX}(K) = \frac{(N-1)Q_{NX}(K) - K}{N-1-K}$$

A resulting value of 0 corresponds to a random embedding, a value of 1 to a perfect embedding of the  $K$ -ary neighborhood.

**Value**

A vector with the values for R\_NX(K)

**Author(s)**

Guido Kraemer

**References**

Lee, J.A., Lee, J.A., Verleysen, M., 2009. Quality assessment of dimensionality reduction: Rank-based criteria. *Neurocomputing* 72.

Lee, J. A., Peluffo-Ordóñez, D. H., & Verleysen, M., 2015. Multi-scale similarities in stochastic neighbour embedding: Reducing dimensionality while preserving both local and global structure. *Neurocomputing*, 169, 246–261. <https://doi.org/10.1016/j.neucom.2014.12.095>

# Index

AUC<sub>ln</sub>K, [3](#)

coRanking (coRanking-package), [2](#)

coranking, [4](#)

coRanking-package, [2](#)

image, [5](#)

imageplot, [5](#)

LCMC, [6](#)

plot\_R\_NX, [6](#)

Q\_NX, [7](#)

R\_NX, [9](#)

rankmatrix, [4](#), [8](#)