

# Package ‘DRquality’

July 21, 2025

**Type** Package

**Title** Quality Measurements for Dimensionality Reduction

**Version** 0.2.1

**Date** 2023-10-10

**Maintainer** Michael Thrun <m.thrun@gmx.net>

**Description** Several quality measurements for investigating the performance of dimensionality reduction methods are provided here. In addition a new quality measurement called Gabriel classification error is made accessible, which was published in Thrun, M. C., Märte, J., & Stier, Q: ``Analyzing Quality Measurements for Dimensionality Reduction" (2023), Machine Learning and Knowledge Extraction (MAKE), <[DOI:10.3390/make5030056](https://doi.org/10.3390/make5030056)>.

**License** GPL-3

**Imports** DatabionicSwarm

**Suggests** plotly, geometry, deldir, FCPS, ProjectionBasedClustering (>= 1.2.1), DataVisualizations, FastKNN, ggplot2, pcaPP,pracma, spdep, grid, igraph, cccd, sf

**Encoding** UTF-8

**NeedsCompilation** no

**Author** Quirin Stier [aut],  
Florian Lerch [ctb],  
Julian Märte [aut],  
Hermann Tafo [ctb],  
Laukert Schlichting [ctb],  
Michael Thrun [aut, cph, cre] (ORCID:  
<<https://orcid.org/0000-0001-9542-5543>>)

**Repository** CRAN

**Date/Publication** 2023-10-12 09:40:15 UTC

## Contents

ClassificationError . . . . .	2
Cmeasure . . . . .	3

GabrielClassificationError . . . . .	4
KendallsTau . . . . .	6
MeasureTandD . . . . .	7
plotMeasureRAAR . . . . .	8
plotMeasureTandD . . . . .	8
PrecisionAndRecall . . . . .	9
RAAR . . . . .	10
SpearmanError . . . . .	11
SpearmanRho . . . . .	12
TopologicalCorrelation . . . . .	12
ZrehenMeasure4All . . . . .	13

## Index 15

---

ClassificationError    *Classification Error (rate)*

---

### Description

Compares projected points to a given prior classification using knn classifier.

### Usage

```
ClassificationError(OutputDistances,Cls,k=5)
```

### Arguments

OutputDistances	[1:n,1:n] numeric matrix with distance matrix of projected data.
Cls	[1:n] Numeric vector containing class information.
k	number of k nearest neighbors, in Venna 2010 set to 5 (here default)

### Details

Projected points are evaluated by k-nearest neighbor classification accuracy (with  $k = 5$ ), that is, each sample in the visualization is classified by majority vote of its  $k$  nearest neighbors in the visualization, and the classification is compared to the ground truth label. [Venna 2010].

### Value

List with three entries:

Error	Classification Error: 1-Accuracy[1]
Accuracy	Accuracy
KNNCls	[1:n]] cls of knn classifier

### Note

Here, the Outputdistances of the Projected points are used.

**Author(s)**

Michael Thrun

**References**

Venna, J., Peltonen, J., Nybo, K., Aidos, H., and Kaski, S. Information retrieval perspective to non-linear dimensionality reduction for data visualization. *The Journal of Machine Learning Research*, 11, 451-490. (2010)

Gracia, A., Gonzalez, S., Robles, V., and Menasalvas, E. A methodology to compare Dimensionality Reduction algorithms in terms of loss of quality. *Information Sciences*, 270, 1-27. (2014)

**Examples**

```
if(requireNamespace("FCPS")){
  data(Hepta,package="FCPS")
  projection=cmdscale(dist(Hepta$Data), k=2)
  ClassificationError(as.matrix(dist(projection)),Hepta$Cls)
}
```

Cmeasure

*C-Measure subtypes***Description**

Calculate the C-Measure subtypes of minimal path length and minimal wiring

**Usage**

```
Cmeasure(Data, Projection, k = 1)
```

**Arguments**

Data	[1:n,1:d] numerical matrix of points in input space.
Projection	[1:n,1:2] numerical matrix of points in output space.
k	Number of nearest neighbors, both measures set it always to k=1.

**Value**

[1:2] Numerical vector of MinimalPathlength and MinimalWiring values.

**Author(s)**

Michael Thrun

**Examples**

```

if(requireNamespace("FCPS")){
  data(Hepta,package="FCPS")
  projection=cmdscale(dist(Hepta$Data), k=2)
  Cmeasure(Hepta$Data,projection)
}

```

---

GabrielClassificationError

*Gabriel Classification Error (GCE)*

---

**Description**

GCE searches for the k-nearest neighbors of the first gabriel neighbors weighted by the Euclidean Distances of the Inputspace [Thrun et al, 2023]. GCE evaluates these neighbors in the Output space. A low value indicates a better two-dimensional projection of the high-dimensional Input space.

**Usage**

```

GabrielClassificationError(Data,ProjectedPoints,Cls,LC,
  PlotIt=FALSE,Plotter = "native", Colors = NULL,LineColor= 'grey',
  main = "Name of Projection", mainSize = 24,xlab = "X", ylab = "Y", xlim, ylim,
  pch,lwd,Margin=list(t=50,r=0,l=0,b=0))

```

**Arguments**

Data	[1:n,1:d] Numeric matrix with n cases and d variables
ProjectedPoints	[1:n,1:2] Numeric matrix with 2D points in cartesian coordinates
Cls	[1:n] Numeric vector with class labels
LC	Optional, Numeric vector of two values determining grid size of the underlying projection
PlotIt	Optional, Boolean: TRUE/FALSE => Plot/Do not plot (Default: FALSE)
Plotter	Optional, Character with plot technique (native or plotly)
Colors	Optional, Character vector of class colors for points
LineColor	Optional, Character of line color used for edges of graph
main	Optional, Character plot title
mainSize	Optional, Numeric size of plot title
xlab	Optional, Character name of x ax
ylab	Optional, Character name of y ax
xlim	Optional, Numeric vector with two values defining x ax range

ylim	Optional, Numeric vector with two values defining y ax range
pch	Optional, Numeric of point size (graphic parameter)
lwd	Optional, Numeric of linewidth (graphic parameter)
Margin	Optional, Margin of plotly plot

### Details

Gabriel Classification Error (GCE) makes an unbiased evaluation of distance- and density-based structures which might be even non-linear separable. First, GCE utilizes the information provided by a prior classification to assess projected structures. Second, GCE applies the insights drawn from graph theory. Details are described in [Thrun et al, 2023].

### Value

list of several entries containing first the GCE itself as main result followed by further entries which contain potential important information

GCE	Numeric: the 'Gabriel Classification Error'
GCEperPoint	[1:n] unnormalized GCE of each point: $GCE = \text{mean}(GCEperPoint)$
nn	the number of points in a relevant neighborhood: $0.5 * 85\text{percentile}(\text{AnzNN})$
AnzNN	[1:n] the number of points with a Gabriel graph neighborhood
NNdists	[1:n,1:nn] the distances within the relevant neighborhood, 1 for inter cluster distances and 0 for inner cluster distances
HD	[1:nn] $HD = \text{HarmonicDecay}(nn)$ i.e weight function for the NNdists: $GCEperPoint = HD * NNdists$
IsInterDistance	Distances to the nn closest neighbors.
GabrielDists	Distance matrix implied by high dimensional distances and the underlying gabriel (Gabriel) graph
ProjectionGraphError	Plotly object in case, plotly is chosen.

### Author(s)

Michael Thrun, Quirin Stier, Julian Märte

### References

[Thrun et al, 2023] Thrun, M.C, Märte, J., Stier, Q.: Analyzing Quality Measurements for Dimensionality Reduction, Machine Learning and Knowledge Extraction (MAKE), Vol 5., accepted, 2023.

### Examples

```
if(requireNamespace("FCPS")){
  data(Hepta,package="FCPS")
  projection=cmdscales(dist(Hepta$Data), k=2)
  GabrielClassificationError(Hepta$Data,projection,Hepta$Cls)$GCE
}
```

```
if(requireNamespace("FCPS")){
  data(Hepta,package="FCPS")
  projection=cmdscales(dist(Hepta$Data), k=2)
  GabrielClassificationError(Hepta$Data,projection,Hepta$Cls)$GCE
}
```

---

KendallsTau

*Statistical correlation by Kendall*

---

### Description

Calculates the statistical correlation by Kendall. Basically a wrapper to `pcaPP::cor.fk`.

### Usage

```
KendallsTau(InputDists, OutputDists)
```

### Arguments

InputDists      Matrix containing the distances of the first dataset.  
OutputDists      Matrix containing the distances of the second dataset.

### Value

Equivalent to `cor.fk`

### Author(s)

Michael Thrun

### Examples

```
if(requireNamespace("FCPS")){
  data(Hepta,package="FCPS")
  InputDist=dist(Hepta$Data)
  projection=cmdscales(InputDist, k=2)
  KendallsTau(as.matrix(InputDist),as.matrix(dist(projection)))
}
```

---

MeasureTandD

*Trustworthiness and Discontinuity.*

---

### Description

In a trustworthy projection the visualized proximities hold in the original data as well, whereas a continuous projection visualizes all proximities of the original data.

### Usage

```
MeasureTandD(Data, pData, NeighborhoodSize)
```

### Arguments

Data	[1:n,1:d] points in input room with d attributes
pData	[1:n,1:2] projected points in output room, with index,x,y or index,line,column
NeighborhoodSize	Integer - sets the maximum number of neighbors to calculate trustworthiness and continuity for.

### Value

Numeric matrix [1:NeighborhoodSize,1:2] containing the trustworthiness values in the first column and the discontinuity values in the second column.

### Author(s)

Julian Märte

### References

Venna, J., & Kaski, S. (2005, September). Local multidimensional scaling with controlled tradeoff between trustworthiness and continuity. In Proceedings of 5th Workshop on Self-Organizing Maps (pp. 695-702).

Kaski, S., Nikkilä, J., Oja, M., Venna, J., Törönen, P., & Castrén, E. (2003). Trustworthiness and metrics in visualizing similarity of gene expression. *BMC bioinformatics*, 4(1), 1-13.

### Examples

```
if(requireNamespace("FCPS")){  
  data(Hepta,package="FCPS")  
  projection=cmdscale(dist(Hepta$Data), k=2)  
  MeasureTandD(Hepta$Data,projection, 2)  
}
```

---

plotMeasureRAAR      *Computes Rescaled Average Agreement Rate*

---

### Description

Rescaled average agreement rate deduced by the co-ranking matrix from LCMC for various different sizes of the neighborhood.

### Usage

```
plotMeasureRAAR(Raar, label = 'ProjectionMethod',
gPlotList = list(RAARplot = ggplot2::ggplot()), LineType="solid", Shape = 16,
PointsPerE = 10, fancy = FALSE)
```

### Arguments

Raar	Output of RAAR() applied for a projection method.
label	Title of plot.
gPlotList	Settings for ggplot.
LineType	Character - graphic parameter: Line type of ggplot.
Shape	Integer: type of point
PointsPerE	Numeric graphic parameter: Distance between markers on plot line
fancy	Boolean graphic parameter: Some automatic settings for a more appealing plot.

### Value

ggplot object

### Author(s)

Michael Thrun

---

plotMeasureTandD      *Computes rank-based smoothed precision and recall*

---

### Description

Compares the projection in pData with the original data in Data and calculates trustworthiness and continuity of the projection for neighborhood sizes ranging from 1 to the size of the neighborhood.

### Usage

```
plotMeasureTandD(TDmatrix, label = 'ProjectionMethod',
gPlotList = list(TW = ggplot2::ggplot(), DC = ggplot2::ggplot()), LineType = "solid",
Shape = 16, PointsPerE = 16)
```



**Arguments**

TDmatrix	Output of MeasureTundD() applied for a projection method.
label	Title of plot.
gPlotList	Settings for ggplot.
LineType	Character - graphic parameter: Line type of ggplot.
Shape	Integer: type of point
PointsPerE	Numeric graphic parameter: Distance between markers on plot line

**Value**

ggplot object

**Author(s)**

Michael Thrun

---

PrecisionAndRecall      *Precision and Recall.*

---

**Description**

Trade-off between missing similar points versus retrieving dissimilar points.

**Usage**

```
PrecisionAndRecall(Data, pData, NeighborhoodSize = 20)
```

**Arguments**

Data	[1:n,1:d] points in input room with d attributes
pData	[1:n,1:2] projected points in output room, with index,x,y or index,line,column
NeighborhoodSize	Sets the 'effective number of neighbors' used to control the width of the Gaussian, NeRV paper Seite 463 setzt Default auf 20

**Value**

Numeric matrix [1:NeighborhoodSize, 1:2] containing the precision values in the first column and the recall values in the second column of the matrix.

**Author(s)**

Felix Pape

**Examples**

```

if(requireNamespace("FCPS")){
  data(Hepta,package="FCPS")
  projection=cmdscale(dist(Hepta$Data), k=2)
  PrecisionAndRecall(Hepta$Data,projection)
}

```

RAAR

*Rescaled average agreement rate***Description**

Rescaled average agreement rate deduced by the co-ranking matrix from LCMC.

**Usage**

```
RAAR(Data, ProjectedPoints, kmax = nrow(Data) - 2, PlotIt = TRUE)
```

**Arguments**

Data	Matrix containing n cases in rows, d variables in columns or a distance matrix which in this case has to be symmetric
ProjectedPoints	n by OutputDimension matrix containing coordinates of the Projection
kmax	maximum of intervall 1:kmax of k nearest neighbors
PlotIt	Optional: Should the output be plottet. Default: TRUE

**Value**

A list containing:

Raar	Rescaled average agreement rate
Aar	Average agreement rate

**Author(s)**

Michael Thrun

**References**

Lee, J. A., Peluffo-Ordonez, D. H., & Verleysen, M. Multiscale stochastic neighbor embedding: Towards parameter-free dimensionality reduction. Paper presented at the Proceedings of 22st European Symposium on Artificial Neural Networks, Computational Intelligence And Machine Learning (ESANN) (2014).

**Examples**

```

if(requireNamespace("FCPS")){
  data(Hepta,package="FCPS")
  projection=cmdscale(dist(Hepta$Data), k=2)
  RAAR(Hepta$Data,projection,kmax=nrow(Hepta$Data)-2,PlotIt=TRUE)
}

```

---

SpearmanError	<i>Calculates the error of a projection with spearman's rank correlation coefficient.</i>
---------------	-------------------------------------------------------------------------------------------

---

**Description**

Calculates the error of a projection with spearman's rank correlation coefficient.

**Arguments**

VectorOfInputDists(1:n2)  
dissimilarities in Input Space between the n data points in vector form as produced by squareform(Dists(1:n,1:n))

VectorOfOutputDists(1:n2)  
dissimilarities in Output Space between the n data points in vector form as produced by squareform(Dists(1:n,1:n))

**Value**

rho rank correlation coefficient

**Author(s)**

Florian Lerch

**Examples**

```

if(requireNamespace("FCPS")){
  data(Hepta,package="FCPS")
  projection=cmdscale(dist(Hepta$Data), k=2)
  SpearmanError(as.matrix(dist(Hepta$Data)),as.matrix(dist(projection)))
}

```

---

Spearman'sRho	<i>Calculates the error of a projection with spearman's rank correlation coefficient</i>
---------------	------------------------------------------------------------------------------------------

---

**Description**

Calculates the error of a projection with spearman's rank correlation coefficient

**Usage**

```
Spearman'sRho(InputDists, OutputDists)
```

**Arguments**

InputDists	[1:d,1:d] numeric matrix with input distances
OutputDists	[1:d,1:d] numeric matrix with output distances

**Value**

rho

**Author(s)**

Julian Märte

**Examples**

```
if(requireNamespace("FCPS")){
  data(Hepta,package="FCPS")
  projection=cmdscale(dist(Hepta$Data), k=2)
  Spearman'sRho(as.matrix(dist(Hepta$Data)),as.matrix(dist(projection)))
}
```

---

TopologicalCorrelation	<i>Topological Correlation</i>
------------------------	--------------------------------

---

**Description**

Calculates the Topological Correlation

**Usage**

```
TopologicalCorrelation(Data,ProjectedPoints,type='norm',method,Kn=0)
```

**Arguments**

Data	[1:n, 1:d] a numeric matrix of the given n-dim. points: the rows represent the points and the columns represent the coordinates in the d-dim. space.
ProjectedPoints	[1:n, 1:2] numeric matrix of Projected Points, if missing, method should be set!
method	Determines whether the selected projections method for a given set of d-dim. points is a good choice. Therefor, a result of 1 means the selected projection method is good, and a result value of 0 means that the Visualization of the given Data in the two-dim. space doesnt fit the problem.
type	How the paths in the adjacency matrix should be weighted. 'norm' representes path lenthgs of 1 and eucldeidean represents the distance in the euclidean metric.
Kn	k nearest neighbours in the graph. only needed in method is isomap and LocallyLinearEmbedding

**Value**

TC value

**Author(s)**

Hermann Tafo, Laukert Schlichting 07/2015

**Examples**

```
#requires DatabionicSwarm v2.2.1

if(requireNamespace("FCPS")){
#data(Hepta,package="FCPS")
#projection=cmdscale(dist(Hepta$Data), k=2)
#TopologicalCorrelation(Hepta$Data,projection)
}
```

---

ZrehenMeasure4All

*ZrehenMeasure4All*

---

**Description**

A generalized version of the Zrehen-measure which defines the neighbourhood with a Gabriel Graph and is therefore not restricted to grid-based projections.

**Usage**

```
ZrehenMeasure4All(Data, Projection, width, height, isToroid = FALSE,
isGrid = TRUE, plotGabriel = FALSE)
```

**Arguments**

Data	[1:n,1:d] points in input room with d attributes
Projection	[1:n,1:2] projected points in output room, with index,x,y or index,line,column
width	Numeric: only necessary if toroid
height	Numeric: only necessary if toroid
isToroid	Boolean: are the points toroid?
isGrid	Boolean: is the grid a toroid?
plotGabriel	Boolean: plot the generated GabrielGraph (TRUE) or not (FALSE). Default: plotGabriel=FALSE.

**Value**

List with	
V\$zrehen	the raw zrehen measure
V\$normedzrehen	the zrehen measure normed by the number of neighbours
v\$neighbourcounter	the number of possible neighbours by which the zrehen measure is normed

**Author(s)**

Florian Lerch 07/2015

**Examples**

```
if(requireNamespace("FCPS")){
  data(Hepta,package="FCPS")
  projection=cmdscale(dist(Hepta$Data), k=2)
  ZrehenMeasure4All(Hepta$Data,projection)$zrehen
}
```

# Index

- \* **Classification Error**
    - ClassificationError, [2](#)
    - GabrielClassificationError, [4](#)
  - \* **ClassificationErrorRate**
    - ClassificationError, [2](#)
  - \* **ClassificationError**
    - ClassificationError, [2](#)
  - \* **Evaluation of projection methods**
    - GabrielClassificationError, [4](#)
  - \* **GCE**
    - GabrielClassificationError, [4](#)
  - \* **Gabriel Classification Error**
    - GabrielClassificationError, [4](#)
  - \* **QM**
    - GabrielClassificationError, [4](#)
  - \* **Quality measurement**
    - GabrielClassificationError, [4](#)
  - \* **Quality measure**
    - GabrielClassificationError, [4](#)
- ClassificationError, [2](#)  
Cmeasure, [3](#)  
cor.fk, [6](#)
- GabrielClassificationError, [4](#)
- KendallsTau, [6](#)
- MeasureTandD, [7](#)
- plotMeasureRAAR, [8](#)  
plotMeasureTandD, [8](#)  
PrecisionAndRecall, [9](#)
- RAAR, [10](#)
- SpearmanError, [11](#)  
SpearmanRho, [12](#)
- TopologicalCorrelation, [12](#)
- ZrehenMeasure4All, [13](#)