

# Package: aricode (via r-universe)

August 25, 2024

**Type** Package

**Title** Efficient Computations of Standard Clustering Comparison Measures

**Version** 1.0.3

**Maintainer** Julien Chiquet <julien.chiquet@inrae.fr>

**Description** Implements an efficient  $O(n)$  algorithm based on bucket-sorting for fast computation of standard clustering comparison measures. Available measures include adjusted Rand index (ARI), normalized information distance (NID), normalized mutual information (NMI), adjusted mutual information (AMI), normalized variation information (NVI) and entropy, as described in Vinh et al (2009) <doi:10.1145/1553374.1553511>. Include AMI (Adjusted Mutual Information) since version 0.1.2, a modified version of ARI (MARI), as described in Sundqvist et al. <doi:10.1007/s00180-022-01230-7> and simple Chi-square distance since version 1.0.0.

**License** GPL (>=3)

**URL** <https://github.com/jchiquet/aricode>

**BugReports** <https://github.com/jchiquet/aricode/issues>

**Encoding** UTF-8

**Imports** Matrix, Rcpp

**Suggests** testthat, spelling

**LinkingTo** Rcpp

**RoxygenNote** 7.2.3

**Language** en-US

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

**RemoteUrl** <https://github.com/jchiquet/aricode>

**RemoteRef** HEAD

**RemoteSha** ffc98928afc202338481d9fbb53f7527bdad28da

## Contents

AMI	2
ARI	3
Chi2	3
clustComp	4
entropy	5
MARI	5
MARIrav	6
NID	7
NMI	7
NVI	8
RI	9
sortPairs	10
<b>Index</b>	<b>11</b>

---

AMI *Adjusted Mutual Information*

---

### Description

A function to compute the adjusted mutual information between two classifications

### Usage

```
AMI(c1, c2)
```

### Arguments

**c1** a vector containing the labels of the first classification. Must be a vector of characters, integers, numerics, or a factor, but not a list.

**c2** a vector containing the labels of the second classification.

### Value

a scalar with the adjusted rand index.

### See Also

[ARI](#), [RI](#), [NID](#), [NVI](#), [NMI](#), [clustComp](#)

### Examples

```
data(iris)
c1 <- cutree(hclust(dist(iris[,-5])), 4)
AMI(c1, iris$Species)
```

---

ARI *Adjusted Rand Index*

---

**Description**

A function to compute the adjusted rand index between two classifications

**Usage**

```
ARI(c1, c2)
```

**Arguments**

c1 a vector containing the labels of the first classification. Must be a vector of characters, integers, numerics, or a factor, but not a list.

c2 a vector containing the labels of the second classification.

**Value**

a scalar with the adjusted rand index.

**See Also**

[RI](#), [NID](#), [NVI](#), [NMI](#), [clustComp](#)

**Examples**

```
data(iris)
c1 <- cutree(hclust(dist(iris[,-5])), 4)
ARI(c1, iris$Species)
```

---

Chi2 *Chi-square statistics*

---

**Description**

A function to compute the Chi-2 statistics

**Usage**

```
Chi2(c1, c2)
```

**Arguments**

c1 a vector containing the labels of the first classification. Must be a vector of characters, integers, numerics, or a factor, but not a list.

c2 a vector containing the labels of the second classification.

**Value**

a scalar with the chi-square statistics.

**See Also**

[ARI](#), [NID](#), [NVI](#), [NMI](#), [clustComp](#)

**Examples**

```
data(iris)
c1 <- cutree(hclust(dist(iris[,-5])), 4)
Chi2(c1,iris$Species)
```

---

clustComp

*Measures of similarity between two classification*

---

**Description**

A function various measures of similarity between two classifications

**Usage**

```
clustComp(c1, c2)
```

**Arguments**

- c1** a vector containing the labels of the first classification. Must be a vector of characters, integers, numerics, or a factor, but not a list.
- c2** a vector containing the labels of the second classification.

**Value**

a list with the RI, ARI, NMI, NVI and NID.

**See Also**

[RI](#), [NID](#), [NVI](#), [NMI](#), [ARI](#)

**Examples**

```
data(iris)
c1 <- cutree(hclust(dist(iris[,-5])), 4)
clustComp(c1,iris$Species)
```

---

entropy	<i>Entropy</i>
---------	----------------

---

**Description**

A function to compute the empirical entropy for two vectors of classification and the joint entropy

**Usage**

```
entropy(c1, c2)
```

**Arguments**

`c1` a vector containing the labels of the first classification. Must be a vector of characters, integers, numerics, or a factor, but not a list.

`c2` a vector containing the labels of the second classification.

**Value**

a list with the two conditional entropies, the joint entropy and output of `sortPairs`.

**Examples**

```
data(iris)
c1 <- cutree(hclust(dist(iris[,-5])), 4)
entropy(c1, iris$Species)
```

---

MARI	<i>Modified Adjusted Rand Index</i>
------	-------------------------------------

---

**Description**

A function to compute a modified adjusted rand index between two classifications as proposed by Sundqvist et al. in prep, based on a multinomial model.

**Usage**

```
MARI(c1, c2)
```

**Arguments**

`c1` a vector containing the labels of the first classification. Must be a vector of characters, integers, numerics, or a factor, but not a list.

`c2` a vector containing the labels of the second classification.

**Value**

a scalar with the modified ARI.

**See Also**

[ARI](#), [NID](#), [NVI](#), [NMI](#), [clustComp](#)

**Examples**

```
data(iris)
c1 <- cutree(hclust(dist(iris[,-5])), 4)
MARI(c1, iris$Species)
```

---

MARIRaw

*raw Modified Adjusted Rand Index*

---

**Description**

A function to compute a modified adjusted rand index between two classifications as proposed by Sundqvist et al. in prep, based on a multinomial model. Raw means, that the index is not divided by the (maximum - expected) value.

**Usage**

```
MARIRaw(c1, c2)
```

**Arguments**

**c1** a vector containing the labels of the first classification. Must be a vector of characters, integers, numerics, or a factor, but not a list.

**c2** a vector containing the labels of the second classification.

**Value**

a scalar with the modified ARI without the division by the (maximum - expected)

**See Also**

[ARI](#), [NID](#), [NVI](#), [NMI](#), [clustComp](#)

**Examples**

```
data(iris)
c1 <- cutree(hclust(dist(iris[,-5])), 4)
MARIRaw(c1, iris$Species)
```

---

NID *Normalized information distance (NID)*

---

**Description**

A function to compute the NID between two classifications

**Usage**

```
NID(c1, c2)
```

**Arguments**

`c1` a vector containing the labels of the first classification. Must be a vector of characters, integers, numerics, or a factor, but not a list.

`c2` a vector containing the labels of the second classification.

**Value**

a scalar with the normalized information distance .

**See Also**

[RI](#), [NMI](#), [NVI](#), [ARI](#), [clustComp](#)

**Examples**

```
data(iris)
c1 <- cutree(hclust(dist(iris[,-5])), 4)
NID(c1, iris$Species)
```

---

NMI *Normalized mutual information (NMI)*

---

**Description**

A function to compute the NMI between two classifications

**Usage**

```
NMI(c1, c2, variant = c("max", "min", "sqrt", "sum", "joint"))
```

**Arguments**

- `c1` a vector containing the labels of the first classification. Must be a vector of characters, integers, numerics, or a factor, but not a list.
- `c2` a vector containing the labels of the second classification.
- `variant` a string in ("max", "min", "sqrt", "sum", "joint"): different variants of NMI. Default use "max".

**Value**

a scalar with the normalized mutual information .

**See Also**

[RI](#), [NID](#), [NVI](#), [ARI](#), [clustComp](#)

**Examples**

```
data(iris)
c1 <- cutree(hclust(dist(iris[,-5])), 4)
NMI(c1, iris$Species)
```

---

NVI

*Normalized variation of information (NVI)*

---

**Description**

A function to compute the NVI between two classifications

**Usage**

```
NVI(c1, c2)
```

**Arguments**

- `c1` a vector containing the labels of the first classification. Must be a vector of characters, integers, numerics, or a factor, but not a list.
- `c2` a vector containing the labels of the second classification.

**Value**

a scalar with the normalized variation of information.

**See Also**

[RI](#), [NID](#), [NMI](#), [ARI](#), [clustComp](#)



**Examples**

```
data(iris)
c1 <- cutree(hclust(dist(iris[,-5])), 4)
NVI(c1,iris$Species)
```

---

RI

*Rand Index*

---

**Description**

A function to compute the rand index between two classifications

**Usage**

```
RI(c1, c2)
```

**Arguments**

**c1** a vector containing the labels of the first classification. Must be a vector of characters, integers, numerics, or a factor, but not a list.

**c2** a vector containing the labels of the second classification.

**Value**

a scalar with the rand index.

**See Also**

[ARI](#), [NID](#), [NVI](#), [NMI](#), [clustComp](#)

**Examples**

```
data(iris)
c1 <- cutree(hclust(dist(iris[,-5])), 4)
RI(c1,iris$Species)
```

---

`sortPairs`*Sort Pairs*

---

**Description**

A function to sort pairs of integers or factors and identify the pairs

**Usage**

```
sortPairs(c1, c2, spMat = FALSE)
```

**Arguments**

<code>c1</code>	a vector of length $n$ with value between 0 and $N1 < n$
<code>c2</code>	a vector of length $n$ with value between 0 and $N2 < n$
<code>spMat</code>	logical: send back the contingency table as sparsely encoded (cost more than the algorithm itself). Default is FALSE

# Index

AMI, 2

ARI, 2, 3, 4, 6–9

Chi2, 3

clustComp, 2–4, 4, 6–9

entropy, 5

MARI, 5

MARIRaw, 6

NID, 2–4, 6, 7, 8, 9

NMI, 2–4, 6, 7, 7, 8, 9

NVI, 2–4, 6–8, 8, 9

RI, 2–4, 7, 8, 9

sortPairs, 10