Package: mvalpha (via r-universe)

May 16, 2026

Type Package

Title Krippendorff's Alpha for Multi-Valued Data

Version 0.5.0

Description Calculate Krippendorff's alpha for multi-valued data using the methods introduced by Krippendorff and Craggs (2016) <doi:10.1080/19312458.2016.1228863>. Nominal, ordinal, interval, and ratio data types are supported, with options to create bootstrapped estimates of alpha and/or parallelize calculations.

Roxygen list(markdown = TRUE)

License AGPL (>= 3)

Encoding UTF-8

URL https://github.com/therealcfdrake/mvalpha

BugReports https://github.com/therealcfdrake/mvalpha/issues

Depends R (>= 4.2.0)

RoxygenNote 7.3.3

LazyData true

Imports stats, utils, rlang, Rdpack

Suggests parallel

RdMacros Rdpack

Repository https://therealcfdrake.r-universe.dev

Date/Publication 2025-09-26 14:19:56 UTC

RemoteUrl https://github.com/therealcfdrake/mvalpha

RemoteRef HEAD

RemoteSha 130b9bdae8fbf65c14d85f3b499fd9d4dc572f38

Contents

`ex_table3` *Published Examples*

Description

These data represent examples found in the original paper describing the calculation of multi-valued Krippendorff's alpha by Krippendorff and Craggs (2016).

Usage

```
ex_table3

ex_table8a

ex_table8b

ex_table8c

ex_table9a

ex_table9b

ex_table9c
```

Format

Each is a multi-valued nominal set with observers as columns and units as rows.

Source

https://doi.org/10.1080/19312458.2016.1228863

References

Krippendorff K, Craggs R (2016). "The Reliability of Multi-Valued Coding of Data." *Communication Methods and Measures*, **10**(4), 181–198. doi:10.1080/19312458.2016.1228863. https://doi.org/10.1080/19312458.2016.

`mvalpha` *Estimate Multi-Valued Krippendorff's Alpha*

Description

`mvalpha()` calculates Krippendorff's alpha statistic when multi-valued observers are allowed to apply multiple values to an observation.

Usage

```
mvalpha(
  data,
  type = "nominal",
  verbose = TRUE,
  n_boot = NULL,
  parallelize = FALSE,
  cluster_size = NULL
)
```

Arguments

`data`	a data frame containing a list column for each observer. Each row represents an observation unit, and each cell contains a vector of 0 to `w` unique values, where `w` is the number of unique values found in the data set. `NA` values are used to represent missing observations and `NULL` values represent the empty set, `{}`, of responses.
`type`	a string describing the data type of the label set. This can be "nominal", "ordinal", "interval", or "ratio" and is used to select the appropriate distance metric.
`verbose`	a logical value which toggles whether status updates are printed to the console while alpha is being calculated.
`n_boot`	an integer representing the number of bootstrap estimates to calculate for mvDo. The default, `NULL`, will not generate additional estimates.
`parallelize`	a logical value indicating whether to implement parallelization using the `parallel` package.
`cluster_size`	an integer describing the number of cores to allocate to parallelization. If `NULL` and `parallelize=TRUE`, then the maximum number of available cores minus 1 will be used.

Value

An object of class `mvalpha`

References

Krippendorff K, Craggs R (2016). "The Reliability of Multi-Valued Coding of Data." *Communication Methods and Measures*, **10**(4), 181–198. doi:10.1080/19312458.2016.1228863. https://doi.org/10.1080/19312458.2016.

Examples

```
library(mvalpha)

### replicate example from Table 3 in Krippendoff and Craggs (2016) with bootstrapped estimates

# View data
ex_table3

# Estimate alpha
x <- mvalpha(ex_table3, verbose = TRUE, n_boot = 500)

# View result
x

# View the unique values observed in the data
x$values

# View the unique labels used to code the data
x$labels

# Histogram of bootstrapped estimates
hist(x$bootstrap_mvalpha)
```

---

new_mvalpha *Create new mvalpha class object*

Description

Wrapper for creating mvalpha class object.

Usage

```
new_mvalpha(
  mvalpha,
  type,
  mvDo,
  mvDe,
  bootstrap_mvalpha,
  unique_cardinalities,
  units,
  observers,
  labels,
  values,
  values_by_unit,
  dist_CK,
  p_CK,
  data
)
```

Arguments

mvalpha	Multi-valued alpha estimate
type	a string describing the data type of the label set. This can be "nominal", "ordinal", "interval", or "ratio" and is used to select the appropriate distance metric.
mvDo	Observed disagreement
mvDe	Expected disagreement
bootstrap_mvalpha	Bootstrap estimates of mvalpha
unique_cardinalities	Numeric vector of the unique cardinalities observed in the data
units	Names of units
observers	Names of observers
labels	Unique labels used in data
values	Unique values used in data
values_by_unit	Table of values by unit
dist_CK	Distance matrix for label sets C and K
p_CK	Probability matrix for label sets C and K
data	a data frame containing a list column for each observer. Each row represents an observation unit, and each cell contains a vector of 0 to w unique values, where w is the number of unique values found in the data set. NA values are used to represent missing observations and NULL values represent the empty set, {}, of responses.

Value

an mvalpha object

---

print.mvalpha *Print mvalpha class object*

Description

Print generic

Usage

```
## S3 method for class 'mvalpha'
print(x, ...)
```

Arguments

x	mvalpha object
...	additional parameters

Value

invisibly returns the alpha estimate of an mvalpha object

`set_ops` *Efficient Set Operations*

Description

Find the intersection and set difference(s) of two sets all at once and more efficiently than calling `base::intersect()` and `base::setdiff()` separately. Based on this stackoverflow answer https://stackoverflow.com/a/72631719

Usage

```
set_ops(A, B, type)
```

Arguments

`A, B`	sets (vectors) of elements
`type`	a string describing the data type of the label set. This can be "nominal", "ordinal", "interval", or "ratio" and is used to select the appropriate distance metric.

Index