Package 'catekappa'

June 9, 2026

Type Package

Title Design and Analysis of Consistency Tests Based on Kappa Statistic

Version 0.1.1

Description Provides a 'Shiny' application and supporting functions for the design and analysis of consistency tests based on Kappa statistic with categorical responses. Wraps 'irr' and 'kappaSize' packages.

License CC0

Encoding UTF-8

Config/roxygen2/version 8.0.0

Imports bslib, irr, kappaSize, shiny, utils

NeedsCompilation no

Author Gai Zheng [aut, cre],
Xincheng Li [aut],
Yingjie Jiangwang [aut],
Panwei Zhao [aut]

Maintainer Gai Zheng <z2118778229@163.com>

Repository CRAN

Date/Publication 2026-06-09 08:30:08 UTC

Contents

`catekappa-package` *catekappa: Design and Analysis of Categorical Agreement Tests Based on Kappa Statistics*

Description

CATEKAPPA (Categorical Agreement Test Evaluation) provides a Shiny interactive application and supporting functions for the design and analysis of categorical agreement tests.

Details

This package wraps the core functionality of the `irr` and `kappaSize` packages:

- **Design stage**: Use `calc_sample_size_kappa` to calculate sample size, supporting 2–5 categories and 2+ raters.
- **Analysis stage**: Use `analyze_kappa` to compute Cohen's, Fleiss', and Light's Kappa statistics.
- **Interactive app**: Use `run_cate_app` to launch the Shiny interface.

Author(s)

Gai Zheng <z2118778229@163.com> Xincheng Li <lxc409014@qq.com> Yingjie Jiangwang <2312055564@qq.com> Panwei Zhao <1581729526@qq.com>

See Also

`run_cate_app`, `calc_sample_size_kappa`, `analyze_kappa`

`analyze_kappa` *Analyze Agreement Using Kappa Statistics*

Description

Analyze consistency between raters using Cohen's, Fleiss', or Light's Kappa. Wraps functions from the irr package.

Usage

```
analyze_kappa(data, type = "cohen", detail = FALSE)
```

Arguments

`data`	A data frame or matrix with subjects as rows and raters as columns.
`type`	Type of kappa: "cohen" (2 raters), "fleiss" (3+ raters), or "light" (3+ raters, pairwise).
`detail`	Logical. If TRUE, returns detailed output including individual scores.

Value

A list with kappa results, interpretation, and data summary.

Examples

```
data <- data.frame(
  Rater1 = c("Yes", "No", "Yes", "Yes", "No"),
  Rater2 = c("Yes", "No", "Yes", "No", "No")
)
analyze_kappa(data, type = "cohen")
```

`calc_sample_size_kappa`

Calculate Sample Size for Kappa Statistic

Description

Calculate required sample size for consistency tests. Directly wraps kappaSize::PowerBinary / Power3Cats / Power4Cats / Power5Cats.

Usage

```
calc_sample_size_kappa(
  kappa0 = 0.4,
  kappa1 = 0.6,
  props = c(0.5, 0.5),
  alpha = 0.05,
  power = 0.8,
  raters = 2
)
```

Arguments

`kappa0`	Null hypothesis value of kappa (H0).
`kappa1`	Alternative hypothesis value of kappa (H1).
`props`	Expected proportions of categories. Must sum to 1.
`alpha`	Significance level. Default 0.05.
`power`	Desired power. Default 0.8.
`raters`	Number of raters (>=2). Default 2.

Value

A list with sample size n and parameters.

Examples

```
calc_sample_size_kappa(kappa0 = 0.4, kappa1 = 0.6, props = c(0.5, 0.5))
calc_sample_size_kappa(kappa0 = 0.4, kappa1 = 0.6,
                       props = c(0.6, 0.3, 0.1), raters = 3)
```

---

interpret_kappa *Interpret Kappa Value*

Description

Interpret the strength of agreement based on Landis and Koch criteria.

Usage

```
interpret_kappa(kappa)
```

Arguments

kappa	Numeric value of Kappa statistic.

Value

A named list with level, description, and color code.

Examples

```
interpret_kappa(0.3)
interpret_kappa(0.75)
```

---

kappa_fixed_n *Fixed N Analysis for Kappa Statistic*

Description

Given a fixed sample size, estimate the lower confidence bound. Wraps kappaSize::FixedNBinary / FixedN3Cats / FixedN4Cats / FixedN5Cats.

Usage

```
kappa_fixed_n(n, kappa0 = 0.4, props = c(0.5, 0.5), alpha = 0.05, raters = 2)
```

Arguments

n	Sample size.
kappa0	Anticipated value of kappa.
props	Category proportions.
alpha	Significance level.
raters	Number of raters.

Value

List with kappaSize raw result and parameters.

---

print.cate_analysis *Print Method for cate_analysis Objects*

Description

Print Method for cate_analysis Objects

Usage

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

Arguments

x	An object of class cate_analysis.
...	Additional arguments.

---

print.cate_design *Print Method for cate_design Objects*

Description

Print Method for cate_design Objects

Usage

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

Arguments

x	An object of class cate_design.
...	Additional arguments.

print.cate_fixed_n *Print Method for cate_fixed_n Objects*

Description

Print Method for cate_fixed_n Objects

Usage

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

Arguments

x	An object of class cate_fixed_n.
...	Additional arguments.

run_cate_app *Run CATE Shiny Application*

Description

Launch the Shiny application for design and analysis of consistency tests based on Kappa statistic with categorical responses.

Usage

```
run_cate_app(
  port = getOption("shiny.port"),
  launch.browser = getOption("shiny.launch.browser", interactive()),
  host = getOption("shiny.host", "127.0.0.1")
)
```

Arguments

port	The TCP port for the application. Defaults to random available port.
launch.browser	Logical. Whether to launch browser automatically.
host	The IPv4 address to listen on.

Value

A Shiny application object (invisible).

Examples

```
if(interactive()){
run_cate_app()
}
```

`summary.cate_analysis` *Summary Method for cate_analysis Objects*

Description

Summary Method for cate_analysis Objects

Usage

```
## S3 method for class 'cate_analysis'
summary(object, ...)
```

Arguments

`object`	An object of class cate_analysis.
`...`	Additional arguments.

Index