Package 'Ridit'

February 19, 2015

Type Package

Title Ridit Analysis (An extension of the Kruskal-Wallis Test.)

Version 1.1

Date 2012-10-15

Author SeyedMahmood TaghaviShahri

Maintainer SeyedMahmood TaghaviShahri <taghavi_m@razi.tums.ac.ir>

Description An extension of the Kruskal-Wallis Test that allow
selection of arbitrary reference group. Also provide Mean Ridit
for each group. Mean Ridit of a group is an estimate of
probability a random observation from that group will be
greater than or equal to a random observation from reference
group.

License GPL-2 | GPL-3

Repository CRAN

Date/Publication 2012-10-15 20:03:34

NeedsCompilation no

R topics documented:

Ridit-package   *Ridit Analysis (An extension of the Kruskal-Wallis Test.)*

Description

An extension of the Kruskal-Wallis Test that allow selection of arbitrary reference group. Also provide Mean Ridit for each group. Mean Ridit of a group is an estimate of probability a random observation from that group will be greater than or equal to a random observation from reference group.

Details

Package:	Ridit
Type:	Package
Version:	1.1
Date:	2012-10-15
License:	GPL-2	GPL-3

Author(s)

SeyedMahmood TaghaviShahri Maintainer: SeyedMahmood TaghaviShahri <taghavi_m@razi.tums.ac.ir>

References

Fleiss, J. L., (1986), The Design and Analysis of Clinical Experiments. New York: John Wiley & Sons.

---

ict   *Inverse CrossTabualtion*

Description

Inverse CrossTabualtion

Usage

```
ict(crosstab)
```

Arguments

crosstab

Author(s)

SeyedMahmood TaghaviShahri

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (crosstab)
{
    crosstab = as.matrix(crosstab)
    n1 = 1:nrow(crosstab)
    n2 = 1:ncol(crosstab)
    d1 = c()
    d2 = c()
    for (i in n1) for (j in n2) {
        count = crosstab[i, j]
        if (is.na(count))
            warning(paste("NA in row", i, ", column", j, "exist!"),
                immediate. = TRUE)
        else if (count > 0)
            for (k in 1:count) {
                d1 = c(d1, n1[i])
                d2 = c(d2, n2[j])
            }
    }
    d1 = factor(d1, labels = rownames(crosstab))
    d2 = factor(d2, labels = colnames(crosstab))
    list(d1 = d1, d2 = d2)
  }
```

---

print.ridit   *Print Output of Ridit analysis*

Description

Print Output of Ridit analysis

Usage

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

Arguments

x	a numeric vector of data values or a matrix of crosstab data.
g	a vector giving group of data or when x is a crosstab, number 1 or 2 when group is in the row or column of crosstab.
...	a optional text corresponds to label or code of arbitrary reference group or a number corresponds to row of group in output (when we want change reference group of output). Also user can enter an arbitrary numeric vector as reference group. Default is Null that used for total of all group as reference (special case that equivalent to the Kruskal-Wallis test).

Author(s)

SeyedMahmood TaghaviShahri

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x, ...)
{
    cat("\n")
    cat("Ridit Analysis:")
    cat("\n")
    cat("\n")
    m = max(nchar(names(x$MeanRidit)))
    cutpoint = 40
    if (m > cutpoint)
        m = cutpoint
    cat("Group", "\t", format("Label", width = m), "\t", "Mean Ridit",
        "\n", sep = "")
    cat("-----", "\t", format("-----", width = m), "\t", "----------",
        "\n", sep = "")
    for (k in 1:length(x$MeanRidit)) cat(k, "\t", format(substr(names(x$MeanRidit)[k],
        start = 1, stop = m), width = m), "\t", round(x$MeanRidit,
        4)[k], "\n", sep = "")
    cat("\n")
    cat(x$msg, "\n")
    cat("chi-squared = ", round(x$TestStatistic, 4), sep = "")
    cat(", df = ", x$df, sep = "")
    cat(", p-value = ", format(x$Sig, digits = 4), sep = "")
    cat("\n")
    cat("\n")
  }
```

---

ridit   *Ridit Analysis*

Description

An extension of the Kruskal-Wallis test that allow specify arbitrary reference group. Also provide Mean Ridit for each group. Mean Ridit of a group is an estimate of probability a random observation from that group will be greater than or equal to a random observation from reference group.

Usage

```
ridit(x, g, ref = NULL)
```

Arguments

x   a numeric vector of data values or a matrix of crosstab data.

g   a vector giving group of data or when x is a crosstab, number 1 or 2 when group is in the row or column of crosstab.

ref   a text corresponds to label or code of arbitrary reference group or a number corresponds to row of group in output (when we want change reference group of output). Also user can enter an arbitrary numeric vector as reference group. Default is Null that used for total of all group as reference (special case that equivalent to the Kruskal-Wallis test).

Author(s)

SeyedMahmood TaghaviShahri

References

Fleiss, J. L., (1986), The Design and Analysis of Clinical Experiments. New York: John Wiley & Sons.

See Also

Special case of Ridit Analysis is `kruskal.test` when reference is total of all groups.

Examples

```
x=airquality$Ozone
g=airquality$Month
kruskal.test(x,g)
ridit(x,g)
ridit(x,g,"5")
ridit(x,g,5)
```

ridit.raw   *Engine function that compute Ridit analysis*

Description

Engine function that compute Ridit analysis

Usage

```
ridit.raw(x, g, ref = NULL)
```

Arguments

x

g

ref

Author(s)

SeyedMahmood TaghaviShahri

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (x, g, ref = NULL)
{
    x = as.numeric(x)
    x = as.vector(x)
    g = as.factor(g)
    levels = levels(g)
    levels(g) = 1:length(levels)
    g = as.vector(g)
    g = as.character(g)
    code = is.numeric(ref)
    ref = as.vector(ref)
    ref = as.character(ref)
    if (length(ref) > 1) {
        x = c(x, ref)
        g = c(g, rep(".ref", length(ref)))
        levels = c(".ref", levels)
    }
    crosstab = t(as.matrix(table(x, g)))
    rownames(crosstab) = levels
    refindex = NULL
    if (length(ref) == 1) {
```

```
        if (!code)
            refindex = which(levels == ref)
        if (code && ref >= 1 && ref <= nrow(crosstab))
            refindex = as.numeric(ref)
    }
    else if (length(ref) > 1)
        refindex = which(levels == ".ref")
    if (length(refindex) != 0)
        refrow = crosstab[refindex, ]
    else refrow = apply(crosstab, 2, sum)
    if (length(refindex) == 0)
        msg = paste("Reference: Total of all groups", sep = "")
    else msg = paste("Reference: Group = ", refindex, ", Label = ",
        levels[refindex], sep = "")
    nref = sum(refrow)
    ridit = 0.5 * refrow[1]/nref
    for (i in 2:length(refrow)) {
        iridit = (sum(refrow[1:i - 1]) + 0.5 * refrow[i])/nref
        ridit = c(ridit, iridit)
    }
    n = apply(crosstab, 1, sum)
    meanRidit = c()
    for (i in 1:nrow(crosstab)) {
        itable = crosstab[i, ]
        meanRidit = c(meanRidit, sum(ridit * itable)/n[i])
    }
    n0 = sum(n)
    rbar0 = sum(n * meanRidit)/n0
    t = apply(crosstab, 2, sum)
    f = 1 - (sum(t * (t - 1) * (t + 1)))/(n0 * (n0 - 1) * (n0 +
        1))
    teststatistic = (12 * n0 * sum(n * (meanRidit - rbar0)^2))/((n0 +
        1) * f)
    testdf = nrow(crosstab) - 1
    pvalue = pchisq(q = teststatistic, df = testdf, lower.tail = FALSE)
    if (length(ref) == 0)
        ref = NULL
    names(meanRidit) = rownames(crosstab)
    output = list(MeanRidit = meanRidit, TestStatistic = teststatistic,
        df = testdf, Sig = pvalue, x = x, g = g, ref = ref, crosstab = crosstab,
        msg = msg)
    class(output) <- c("ridit", class(output))
    output
}
```

Index