

# Package ‘MCPtests’

May 7, 2026

**Type** Package

**Title** Multiples Comparisons Procedures

**Version** 1.0.1

**Description** Performs the execution of the main procedures of multiple comparisons in the literature, Scott-Knott (1974) <<http://www.jstor.org/stable/2529204>>, Batista (2016) <<http://repositorio.ufla.br/jspui/handle/1/11466>>, including graphic representations and export to different extensions of its results. An additional part of the package is the presence of the performance evaluation of the tests (Type I error per experiment and the power). This will assist the user in making the decision for the chosen test.

**Imports** stats, base, utils, graphics, grDevices, SMR, writexl, xtable, tcltk, foreach, doParallel, parallel

**Suggests** tkrplot

**License** GPL (>= 2)

**LazyData** TRUE

**RoxygenNote** 7.1.1

**Depends** R (>= 2.10)

**NeedsCompilation** no

**Author** Ben Deivide [aut, cre],  
Daniel Furtado [aut],  
Diego Arthur [aut],  
Matheus Fernando [ctb]

**Maintainer** Ben Deivide <[ben.deivide@ufsj.edu.br](mailto:ben.deivide@ufsj.edu.br)>

**Repository** CRAN

**Date/Publication** 2020-12-17 16:30:17 UTC

## Contents

ex1ScottKnott . . . . .	2
guiMCP . . . . .	2

MCPbarplot . . . . .	3
MCPtest . . . . .	4
MCPwrite . . . . .	7

<b>Index</b>	<b>9</b>
--------------	----------

---

ex1ScottKnott	<i>Barley grain yields in bushels per acre</i>
---------------	--

---

### Description

Set of results presented by Duncan (1955), in which Scott-Knott (1974) presented an application of the development of the Scott-Knott test in a randomized block design. References: 1) SCOTT, A. J.; KNOTT, M. A cluster analysis method for grouping means in the analysis of variance. *Biometrics*, International Biometric Society, v. 30, n. 3, p. 507-512, 1974. 2) DUNCAN, D. B. Multiple range and multiple F Tests. *Biometrics*, International Biometric Society, v. 11, n. 1, p. 1-42, 1955.

### Usage

ex1ScottKnott

### Format

A list with 4 variables:

**treatment** Seven varieties of barley conducted by E. Shulkcum of this Institute at Accomc, Virginia, in 1951

**averages** Averages of the seven varieties of barley

**mserror** Result of mean squared error of the analysis of variance of the experiment

**dferror** Degrees of freedom of error of the experiment

**replication** Number of replications of treatments.

---

guiMCP	<i>Graphical User Interface for MCPtests package</i>
--------	--

---

### Description

guiMCP A Graphical User Interface (GUI) for the MCP package

### Usage

guiMCP(gui = TRUE)

### Arguments

gui Logical argument, TRUE or FALSE. The default is TRUE

**Value**

guiMCP presents a GUI for the results of the multiple comparison procedures in the literature. In addition, the GUI returns a graph of the results, as well as the export of these results to three types of file extension and latex code.

**Examples**

```
# Loading package
library(MCPtests)
if (interactive()) {
  guiMCP(gui = FALSE)
}
```

---

MCPbarplot

*Plotting the result of the multiple comparison procedures*


---

**Description**

MCPbarplot creates a bar plot with vertical or horizontal bars to compare the mean treatments by the tests: means grouping based on midrange test, means grouping based on range test, Student-Newman-Keuls and Tukey based on midrange tests, Scott-Knott's test

**Usage**

```
MCPbarplot(
  x,
  MCP = "all",
  col = grDevices::heat.colors(10),
  horiz = FALSE,
  ...
)
```

**Arguments**

x	An object of the MCPtest function
MCP	Allows choosing the multiple comparison test. The <i>default</i> is "all". This option will perform all tests available in the MCPtest object.
col	A specification for the plotting color. The <i>default</i> is <code>heat.colors(10)</code> .
horiz	a logical value. If FALSE, the bars are drawn vertically with the first bar to the left. If TRUE, the bars are drawn horizontally with the first at the bottom.
...	Parameters of the <a href="#">barplot</a> function

**Details**

The MCP argument allows choosing several tests of multiple comparisons from the MCPtest object. For plots in papers, use `col = gray.colors(10)`. For details, see [colors](#) function.

**Value**

MCPbarplot return the bar plot of the tests chosen ("MGM", "MGR", "SNKM", "TM", "SK") to evaluate the treatment means.

**Examples**

```
# Simulated data (completely randomized design)

rv <- c(100.08, 105.66, 97.64, 100.11, 102.60, 121.29, 100.80,
        99.11, 104.43, 122.18, 119.49, 124.37, 123.19, 134.16,
        125.67, 128.88, 148.07, 134.27, 151.53, 127.31)

# Treatments
treat <- factor(rep(LETTERS[1:5], each = 4))

# Anova
res <- aov(rv~treat)

# Loading the MCPtests package
library(MCPtests)

# Choosing tests
results <- MCPtest(y = res, trt = "treat", alpha = 0.05,
                  main = "Multiple Comparison Procedures",
                  MCP = c("MGM", "TM"))

MCPbarplot(results, MCP = "all") # It will be shown two
                                # graphs. First, for the
                                # results of \code{'MGM'}
                                # and the second for the
                                # results of \code{'TM'}.

MCPbarplot(results, MCP = "MGM") # It will be shown
                                  # only the graph
                                  # for the result of
                                  # \code{'MGM'}

# Plot for papers
MCPbarplot(results, MCP = "all", col = gray.colors(10))
```

---

MCPtest

---

*Multiple comparison procedures to the means.*


---

**Description**

MCPtest applies several tests of multiple comparisons present in the literature. The tests chosen are based on the evaluation of the researcher to decide the choice of test for analysis in the experiment.

**Usage**

```
MCPtest(
  y,
  trt = NULL,
  dferror = NULL,
  mserror = NULL,
  replication = NULL,
  alpha = 0.05,
  main = NULL,
  MCP = "all",
  ismean = FALSE,
  parallel = FALSE
)
```

**Arguments**

y	Model (aov or lm), numeric vector containing the response variable or the mean of the treatments.
trt	Constant (y = model) or a vector containing the treatments.
dferror	Degrees of freedom of the Mean Square Error.
mserror	Mean Square Error.
replication	Number de repetitions of the treatments in the experiment. For unbalanced data should be informed the harmonic mean of repetitions. This argument should be informed only if ismean = TRUE.
alpha	Significant level. The default is alpha = 0.05.
main	Title of the analysis.
MCP	Allows choosing the multiple comparison test; the <i>default</i> is "all". This option will go perform all tests. However, the options are: the Skott-Knott midrange test ("MGM"), the Skott-Knott Range test ("MGR"), the Tukey midrange test ("TM"), the Scott-Knott's test ("SK").
ismean	Logic. If FALSE (default), the y argument represents a model (aov or lm) or a numeric vector containing the response variable. If TRUE the y argument represents the mean of treatments.
parallel	Logic. If FALSE (default), the MCPtest function runs without parallelization. Otherwise, the code is executed with parallelization. Note that the parallelization is not always more efficient.

**Details**

The MCP argument allows you to choose various tests of multiple comparisons at once. For example, MCP = c("MGM", "MGR"), and so on.

**Value**

MCPtest returns the print of a list of results. First, the summary of y. Second, the statistics of the test chosen. And finally, the mean group results for each test. If MCPtest function is stored in an object, the results will be printed and also stored in the object.

## References

BATISTA, Ben Deivide de Oliveira. Testes de comparacoes multiplas baseados na distribuicao da *midrange estudentizada* externamente. 2016. 194f. Tese (Doutorado em Estatistica e Experimentacao Agropecuaria) - Universidade Federal de Lavras, 2016. (Portuguese, Brazil)

SCOTT, A. J.; KNOTT, M. A cluster analysis method for grouping means in the analysis of variance. *Biometrics*, International Biometric Society, v. 30, n. 3, p. 507-512, 1974.

DUNCAN, D. B. Multiple range and multiple F Tests. *Biometrics*, International Biometric Society, v. 11, n. 1, p. 1-42, 1955.

## Examples

```
# Simulated data (completely randomized design)

# Response variable
rv <- c(100.08, 105.66, 97.64, 100.11, 102.60, 121.29, 100.80,
        99.11, 104.43, 122.18, 119.49, 124.37, 123.19, 134.16,
        125.67, 128.88, 148.07, 134.27, 151.53, 127.31)

# Treatments
treat <- factor(rep(LETTERS[1:5], each = 4))

# Anova
res <- anova(aov(rv~treat))
DFerror <- res$Df[2]
MSerror <- res$`Mean Sq`[2]

# Loading the MCPtests package
library(MCPtests)

# applying the tests
results <- MCPtest(y = rv,
                  trt = treat,
                  dferror = DFerror,
                  mserror = MSerror,
                  alpha = 0.05,
                  main = "Multiple Comparison Procedure: MGM test",
                  MCP = c("MGM"))

# Other option for the MCP argument is "all". All tests are used.

results$Groups      # Results of the tests
results$Statistics  # Main arguments of the tests
results$Summary     # Summary of the response variable

# Using the y argument as aov or lm model
res <- aov(rv~treat)

MCPtest(y = res, trt = "treat", alpha = 0.05,
        main = "Multiple Comparison Procedure: MGM test",
        MCP = c("MGM"))
```

```

# For unbalanced data: It will be used the harmonic mean of
# the number of experiment replicates

# Using the previous example
rv <- rv[-1]
treat <- treat[-1]

res <- lm(rv~treat) # Linear model

# Multiple comparison procedure: MGR test
MCPtest(y = res, trt = "treat", alpha = 0.05,
        main = "Multiple Comparison Procedure: MGR test",
        MCP = c("MGR"))

# Assuming that the available data are the averages
# of the treatments and the analysis of variance

# Analysis of Variance Table

# Response: rv
#           Df Sum Sq Mean Sq F value    Pr(>F)
# treat      4 4135.2  1033.80  14.669 4.562e-05 ***
# Residuals 15 1057.1    70.47

mean.treat <- c(100.87, 105.95, 117.62, 127.97, 140.30)
treat <- factor(LETTERS[1:5])
DFerror <- 15
MSerror <- 70.47488
replic <- 4

MCPtest(y = mean.treat,
        trt = treat,
        dferror = DFerror,
        mserror = MSerror,
        replication = replic,
        alpha = 0.05,
        main = "Multiple Comparison Procedure: MGM test",
        MCP = c("MGM"),
        ismean = TRUE)

```

---

MCPwrite

*Export the results of the MCPtest function.*


---

## Description

The x object from a MCPtest is written to file arguments.

## Usage

```
MCPwrite(x, MCP = "all", extension = "csv", dataMR = "all")
```

**Arguments**

x	object from the MCPtest functions.
MCP	Allows choosing the multiple comparison test. The <i>default</i> is "all". This option will go perform all tests from the MCPtest object.
extension	Type of format of the file. Four options "csv", "txt", "xlsx" and "latex". The <i>default</i> is "csv".
dataMR	Allows to choose the results to be written. Three options are available: "groups", "summary" or "all". The option "groups" writes the treatment mean groups evaluated by the chosen test in the MCP argument. The "summary" writes the descriptive statistics of the response variable. The options "all" should be chosen for both results.

**Details**

Note that the choice of the tests in the MCPwrite function must be in accordance with the tests chosen in the x argument.

**Value**

MCPwrite writes the most important results for the chosen tests in the MCP argument.

**Examples**

```
# Simulated data (completely randomized design)

rv <- c(100.08, 105.66, 97.64, 100.11, 102.60, 121.29, 100.80,
        99.11, 104.43, 122.18, 119.49, 124.37, 123.19, 134.16,
        125.67, 128.88, 148.07, 134.27, 151.53, 127.31)

# Treatments
treat <- factor(rep(LETTERS[1:5], each = 4))

# Anova
res <- aov(rv~treat)

# Loading the MCPtests package
library(MCPtests)

# Choosing any tests
results <- MCPtest(y = res, trt = "treat", alpha = 0.05,
                  main = "Multiple Comparison Procedures",
                  MCP = c("MGM", "TM"))

#Export file in latex (Output in Console)
MCPwrite(results, MCP = "all", extension = "latex", dataMR = "all")

#Observation: The MCPwrite function export
#              only one extension at a time
```

# Index

## \* datasets

ex1ScottKnott, [2](#)

barplot, [3](#)

colors, [3](#)

ex1ScottKnott, [2](#)

guiMCP, [2](#)

MCPbarplot, [3](#)

MCPtest, [4](#)

MCPwrite, [7](#)