Package: MarginalMaxTest (via r-universe)

October 18, 2026

Title Max-Type Test for Marginal Correlation with Bootstrap

Version 1.0.1

Description Test the marginal correlation between a scalar response variable with a vector of explanatory variables using the max-type test with bootstrap. The test is based on the max-type statistic and its asymptotic distribution under the null hypothesis of no marginal correlation. The bootstrap procedure is used to approximate the null distribution of the test statistic. The package provides a function for performing the test. For more technical details, refer to Zhang and Laber (2014) <doi:10.1080/01621459.2015.1106403>.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.3

LinkingTo Rcpp, RcppArmadillo

Imports Rcpp

URL https://github.com/canyi-chen/MarginalMaxTest

BugReports https://github.com/canyi-chen/MarginalMaxTest/issues

NeedsCompilation yes

Author Canyi Chen [aut, cre, cph]
(<https://orcid.org/0000-0002-0673-5812>)

Maintainer Canyi Chen <cychen.stats@outlook.com>

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

Date/Publication 2025-04-23 10:41:36 UTC

RemoteUrl https://github.com/cran/MarginalMaxTest

RemoteRef HEAD

RemoteSha c22a0bab6f7bd7bc7c6c4d53cad4f4f22ab454a3

Contents

marginal.test 2

Index 3

`marginal.test` *Test for marginal effects of predictors on a scalar response*

Description

Test for marginal effects of predictors on a scalar response

Usage

```
marginal.test(x, y, B = 199L, method = "adaptive")
```

Arguments

`x`	A numeric matrix of predictors
`y`	A numeric vector of responses
`B`	Number of bootstrap samples (default 199)
`method`	Method for p-value calculation: "max", "sum", or "adaptive"

Value

A list containing the p-value and computation time

Note

This function is based on the C implementation by Zhang and Laber (2014) doi:10.1080/01621459.2015.1106403.

Examples

```
# Generate sample data
set.seed(47)
n <- 200
p <- 10
x <- matrix(rnorm(n*p), n, p)
y <- 0.25*x[,1] + rnorm(n)
# Run the test
marginal.test(x, y, B = 200, method = "adaptive")
marginal.test(x, y, B = 200, method = "max")
marginal.test(x, y, B = 200, method = "sum")
```

Index

marginal.test, 2