

# Package ‘webbioc’

January 24, 2026

**Version** 1.83.0

**Date** 2009-02-05

**Title** Bioconductor Web Interface

**Author** Colin A. Smith <colin@colinsmith.org>

**Maintainer** Colin A. Smith <colin@colinsmith.org>

**Depends** R (>= 1.8.0), Biobase, affy, multtest, annaffy, vsn, gcrma, qvalue

**Imports** multtest, qvalue, stats, utils, BiocManager

**SystemRequirements** Unix, Perl (>= 5.6.0), Netpbm

**Description** An integrated web interface for doing microarray analysis using several of the Bioconductor packages. It is intended to be deployed as a centralized bioinformatics resource for use by many users. (Currently only Affymetrix oligonucleotide analysis is supported.)

**License** GPL (>= 2)

**URL** <http://www.bioconductor.org/>

**LazyLoad** yes

**biocViews** Infrastructure, Microarray, OneChannel, DifferentialExpression

**git\_url** <https://git.bioconductor.org/packages/webbioc>

**git\_branch** devel

**git\_last\_commit** 8bbf95c

**git\_last\_commit\_date** 2025-10-29

**Repository** Bioconductor 3.23

**Date/Publication** 2026-01-23

## Contents

installReps . . . . .	2
mt.wrapper . . . . .	2

**Index****5**


---

installReps	<i>Install all repository packages</i>
-------------	--

---

**Description**

Using reposTools, install/update all packages from given repositories.

**Usage**

```
installReps(repNames = "aData", lib = .libPaths()[1],
            type = getOption("pkgType"))
```

**Arguments**

repNames	A character vector containing repository names. A listing of known repository names can be found with the getReposOption() function.
lib	A path to install/update the packages. If this directory does not exist, it will be created (if possible).
type	type of package do download

**Author(s)**

Colin A. Smith <webbioc@colinsmith.org>

---

mt.wrapper	<i>Multiple Testing Wrapper Function</i>
------------	--

---

**Description**

A wrapper for some of the functionality for the multtest package. It also includes hooks to calculate q-values with John D. Storey's 'q-value.R' code.

**Usage**

```
mt.wrapper(proc, X, classlabel, test="t", rawpcalc="Parametric", side="abs", ...)
```

**Arguments**

<code>proc</code>	A character string containing the name of the multiple testing procedure for which adjusted $p$ -values are to be computed. This vector should include any of the following: "Bonferroni", "Holm", "Hochberg", "SidakSS", "SidakSD", "BH", "BY", "maxT", "minP", "q".
<code>X</code>	A data frame or matrix, with $m$ rows corresponding to variables (hypotheses) and $n$ columns to observations. In the case of gene expression data, rows correspond to genes and columns to mRNA samples. The data can be read using <a href="#">read.table</a> .
<code>classlabel</code>	A vector of integers corresponding to observation (column) class labels. For $k$ classes, the labels must be integers between 0 and $k - 1$ . For the blockf test option, observations may be divided into $n/k$ blocks of $k$ observations each. The observations are ordered by block, and within each block, they are labeled using the integers 0 to $k - 1$ .
<code>test</code>	<p>A character string specifying the statistic to be used to test the null hypothesis of no association between the variables and the class labels.</p> <p>If <code>test="t"</code>, the tests are based on two-sample Welch t-statistics (unequal variances).</p> <p>If <code>test="t.equalvar"</code>, the tests are based on two-sample t-statistics with equal variance for the two samples. The square of the t-statistic is equal to an F-statistic for <math>k = 2</math>.</p> <p>If <code>test="wilcoxon"</code>, the tests are based on standardized rank sum Wilcoxon statistics.</p> <p>If <code>test="f"</code>, the tests are based on F-statistics.</p> <p>If <code>test="pairt"</code>, the tests are based on paired t-statistics. The square of the paired t-statistic is equal to a block F-statistic for <math>k = 2</math>.</p> <p>If <code>test="blockf"</code>, the tests are based on F-statistics which adjust for block differences (cf. two-way analysis of variance).</p>
<code>rawpcalc</code>	A character string specifying how to calculate nominal/raw p-values. The possible choices are "Parametric" or "Permutation".
<code>side</code>	<p>A character string specifying the type of rejection region.</p> <p>If <code>side="abs"</code>, two-tailed tests, the null hypothesis is rejected for large absolute values of the test statistic.</p> <p>If <code>side="upper"</code>, one-tailed tests, the null hypothesis is rejected for large values of the test statistic.</p> <p>If <code>side="lower"</code>, one-tailed tests, the null hypothesis is rejected for small values of the test statistic.</p>
<code>...</code>	Further arguments for <code>mt.maxT</code> , <code>mt.minP</code> , <code>qvalue</code> .

**Value**

A data frame with components

<code>index</code>	Vector of row indices, between 1 and <code>nrow(X)</code> , where rows are sorted first according to their adjusted $p$ -values, next their unadjusted $p$ -values, and finally their test statistics.
--------------------	--

<code>teststat</code>	Vector of test statistics, ordered according to <code>index</code> . To get the test statistics in the original data order, use <code>teststat[order(index)]</code> .
<code>rawp</code>	Vector of raw (unadjusted) $p$ -values, ordered according to <code>index</code> .
<code>adjp</code>	Vector of adjusted $p$ -values, ordered according to <code>index</code> .
<code>plower</code>	For <code>mt.minP</code> function only, vector of "adjusted $p$ -values", where ties in the permutation distribution of the successive minima of raw $p$ -values with the observed $p$ -values are counted only once. Note that procedures based on <code>plower</code> do not control the FWER. Comparison of <code>plower</code> and <code>adjp</code> gives an idea of the discreteness of the permutation distribution. Values in <code>plower</code> are ordered according to <code>index</code> .

**Author(s)**

Colin A. Smith <webbioc@colinsmith.org>

# Index

- \* **file**
  - installReps, [2](#)
  - mt.wrapper, [2](#)
- installReps, [2](#)
- mt.minP, [4](#)
- mt.wrapper, [2](#)
- read.table, [3](#)