

Package ‘testthatmulti’

May 8, 2026

Type Package

Title Testing for R Packages with Multiple Attempts for Noisy Tests

Version 0.2.0

Maintainer Collin Erickson <collinberickson@gmail.com>

Description Runs tests using the 'testthat' package but allows for multiple attempts for a single test. This is useful for noisy or flaky tests that generally pass but can fail due to occasional random errors, such as numeric instability or using random data.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.2

URL <https://github.com/CollinErickson/testthatmulti>

BugReports <https://github.com/CollinErickson/testthatmulti/issues>

Suggests testthat (>= 3.0.0), rlang

Config/testthat/edition 3

Language en-US

NeedsCompilation no

Author Collin Erickson [aut, cre]

Repository CRAN

Date/Publication 2025-08-24 23:50:02 UTC

Contents

ttm	2
ttm_expect_equal	3
ttm_expect_error	4
ttm_expect_no_error	5
ttm_expect_true	6
ttm_i	7
ttm_n	7

Index	9
--------------	----------

`ttm`*Test that with multiple attempts*

Description

Test that with multiple attempts

Usage

```
ttm(n, expr, verbose = 0)
```

Arguments

<code>n</code>	Maximum number of attempts
<code>expr</code>	Expression to evaluate
<code>verbose</code>	Amount that should be printed

Value

Nothing

Examples

```
set.seed(0)

# 1 attempt, all pass
ttm(1, {
  ttm_expect_true(TRUE)
  ttm_expect_true(1 == 1)
  ttm_expect_true(all(1:5 == 1:5))
})

# Fails first 10 times, then passes
ttm(100, {
  x <- runif(1)
  print(x)
  ttm_expect_true(x < 0.1)
})

# Will always fail regardless of number of attempts
try({
  ttm(3, {
    ttm_expect_true(1 == 2)
  })
})
```

ttm_expect_equal	<i>Test that multi: expect equal</i>
------------------	--------------------------------------

Description

Test that multi: expect equal

Usage

```
ttm_expect_equal(
  object,
  expected,
  ...,
  tolerance = if (testthat::edition_get() >= 3) testthat::testthat_tolerance(),
  info = NULL,
  label = NULL,
  expected.label = NULL,
  verbose = 0
)
```

Arguments

object	Object to check if equal to expected
expected	Expected value
...	Args passed to testthat::expect_equal()
tolerance	Passed to 'testthat::expect_true()'. Default is 1e-10.
info	Passed to 'testthat::expect_true()'. Default is NULL.
label	Passed to 'testthat::expect_true()'. Default is NULL.
expected.label	Passed to 'testthat::expect_true()'. Default is NULL.
verbose	Amount of info that should be printed. Default is 0.

Value

Test result

Examples

```
set.seed(0)

# 1 attempt, all pass
ttm(1, {
  ttm_expect_equal(TRUE, TRUE)
  ttm_expect_equal(1, 1)
  ttm_expect_equal(1:5, 1:5)
})
```

```

# Fails first 6 times, then passes
ttm(100, {
  x <- sample(1:6, 1)
  print(x)
  ttm_expect_equal(x, 3)
})

# Will always fail regardless of number of attempts
try({
  ttm(3, {
    ttm_expect_equal(1, 2)
  })
})

```

ttm_expect_error	<i>Test that multi: expect error</i>
------------------	--------------------------------------

Description

See ‘testthat::expect_error’ for details.

Usage

```
ttm_expect_error(object, info = NULL, label = NULL, verbose = 0)
```

Arguments

object	Object to test.
info	Passed to ‘testthat::expect_error()’.
label	Passed to ‘testthat::expect_error()’.
verbose	Amount of info that should be printed.

Value

Test result

Examples

```

set.seed(0)

# 1 attempt, all pass
ttm(1, {
  ttm_expect_error(stop('error'))
})

# Fails first ~10 times, then passes
ttm(100, {
  ttm_expect_error({
    print(1)

```

```

    if (runif(1) < 0.1) {
      stop('give error')
    }
  })
})

# Will always fail regardless of number of attempts
try({
  ttm(3, {
    ttm_expect_error(1 == 2)
  })
})

```

ttm_expect_no_error *Test that multi: expect no error*

Description

See ‘testthat::expect_no_error’ for details.

Usage

```
ttm_expect_no_error(object, message = NULL, class = NULL, verbose = 0)
```

Arguments

object	Object to test.
message	Passed to ‘testthat::expect_no_error()’.
class	Passed to ‘testthat::expect_no_error()’.
verbose	Amount of info that should be printed.

Value

Test result

Examples

```

set.seed(0)

# 1 attempt, all pass
ttm(1, {
  ttm_expect_no_error(1)
})

# Fails first ~10 times, then passes
ttm(100, {
  ttm_expect_no_error({
    if (runif(1) > 0.1) {
      print(1)
    }
  })
})

```

```

        stop('give error')
      }
    })
  })

# Will always fail regardless of number of attempts
try({
  ttm(3, {
    ttm_expect_no_error(stop('error'))
  })
})

```

ttm_expect_true	<i>Test that multi: expect true</i>
-----------------	-------------------------------------

Description

See ‘testthat::expect_true’ for details.

Usage

```
ttm_expect_true(object, info = NULL, label = NULL, verbose = 0)
```

Arguments

object	Object to test.
info	Passed to ‘testthat::expect_true()’.
label	Passed to ‘testthat::expect_true()’.
verbose	Amount of info that should be printed.

Value

Test result

Examples

```

set.seed(0)

# 1 attempt, all pass
ttm(1, {
  ttm_expect_true(TRUE)
  ttm_expect_true(1 == 1)
  ttm_expect_true(all(1:5 == 1:5))
})

# Fails first 10 times, then passes
ttm(100, {
  x <- runif(1)
  print(x)
})

```

```
    ttm_expect_true(x < 0.1)
  })

# Will always fail regardless of number of attempts
try({
  ttm(3, {
    ttm_expect_true(1 == 2)
  })
})
```

ttm_i	<i>Iteration count for ttm()</i>
-------	----------------------------------

Description

Iteration count for ttm()

Usage

```
ttm_i()
```

Value

int for ttm iteration, starting at 1

Examples

```
ttm(100, {
  x <- runif(1)
  cat('ttm i =', ttm_i(), 'n =', ttm_n(), 'x =', x, '\n')
  ttm_expect_true(x < 0.1)
})
```

ttm_n	<i>Max number of attempts for ttm()</i>
-------	---

Description

Max number of attempts for ttm()

Usage

```
ttm_n()
```

Value

int for ttm number of attempts

Examples

```
ttm(100, {  
  x <- runif(1)  
  cat('ttm i =', ttm_i(), 'n =', ttm_n(), 'x =', x, '\n')  
  ttm_expect_true(x < 0.1)  
})
```

Index

ttm, [2](#)
ttm_expect_equal, [3](#)
ttm_expect_error, [4](#)
ttm_expect_no_error, [5](#)
ttm_expect_true, [6](#)
ttm_i, [7](#)
ttm_n, [7](#)