

Package: ReliaLearnR (via r-universe)

May 30, 2026

Type Package

Title Learning Modules for Reliability Analysis

Version 0.5

URL <https://paulgovan.github.io/ReliaLearnR/>,
<https://github.com/paulgovan/ReliaLearnR>

BugReports <https://github.com/paulgovan/ReliaLearnR/issues>

Description Learning modules for reliability analysis including modules for Reliability, Availability, and Maintainability (RAM) Analysis, Life Data Analysis, Reliability Testing, Repairable Systems Analysis, and Reliability Block Diagrams.

Depends R (>= 4.1.0)

Imports learnr (>= 0.11.0), ReliaGrowR, WeibullR, WeibullR.ALT

Suggests DiagrammeR, knitr, mockery, ReliaPlotR, ReliaShiny, rmarkdown, testthat (>= 3.0.0)

License Apache License (>= 2)

Encoding UTF-8

RoxygenNote 7.3.3

VignetteBuilder knitr

Config/testthat/edition 3

Config/pak/sysreqs cmake make libicu-dev libsodium-dev libuv1-dev libssl-dev zlib1g-dev

Repository <https://paulgovan.r-universe.dev>

Date/Publication 2026-05-29 23:17:30 UTC

RemoteUrl <https://github.com/paulgovan/relialearnr>

RemoteRef HEAD

RemoteSha 4f32a154f6386d55dd356f5deadf8612f8dd93e2

Contents

avail	2
fr	3
lda	3
mtbf	4
mttf	4
ram	5
rbd	6
rel	6
rs	7
rt	7
Index	9

avail	<i>Availability (1 - unavailability / total)</i>
-------	--

Description

Availability is the proportion of time a system is in a functioning condition. This function computes availability given unavailable times and total times.

Usage

```
avail(unavailTime, totalTime)
```

Arguments

unavailTime Numeric scalar or numeric vector of unavailable times.
totalTime Numeric scalar or numeric vector of total times (same units as unavailTime).

Value

Numeric scalar: availability for the period (between 0 and 1).

Examples

```
avail(100, 1000)
avail(c(5,10), c(500,600))
```

fr	<i>Failure rate (lambda)</i>
----	------------------------------

Description

Failure rate is the frequency with which an engineered system or component fails, expressed in failures per unit of time. This function computes failure rate given failure counts and total operating times.

Usage

```
fr(failures, totalTime)
```

Arguments

failures	Numeric scalar or numeric vector of failure counts (non-negative).
totalTime	Numeric scalar or numeric vector of total operating times.

Value

Numeric scalar: failures per unit time (failures / totalTime).

Examples

```
fr(75, 5000)  
fr(c(10,5), c(1000,2000))
```

lda	<i>Launch the Life Data Analysis Tutorial</i>
-----	---

Description

This function launches an interactive tutorial for life data analysis.

Usage

```
lda()
```

Value

This function does not return a value.

See Also

<https://paulgovan.github.io/ReliaLearnR/>

Examples

```
if (interactive()) {
  lda()
}
```

mtbf

Mean Time Between Failures (MTBF) for repairable items.

Description

MTBF = total operating time / number of failures. The MTBF is the expected time between consecutive failures. It is commonly used for repairable items. The behavior is the same as mttf here; keep separate name for semantic clarity.

Usage

```
mtbf(failures, totalTime)
```

Arguments

failures Numeric scalar or numeric vector of failure counts (non-negative).
totalTime Numeric scalar or numeric vector of total operating times.

Value

Numeric scalar: MTBF. If number of failures is zero, returns Inf (with a warning).

Examples

```
mtbf(5, 1000)
mtbf(c(2,3), c(500,500))
```

mttf

Mean Time To Failure (MTTF)

Description

For non-repairable items MTTF = total operating time / number of failures. The MTTF is the expected time to the first failure. It is commonly used for non-repairable items. The behavior implemented is the same as mtbf here; keep separate name for semantic clarity.

Usage

```
mttf(failures, totalTime)
```

Arguments

`failures` Numeric scalar or numeric vector of failure counts (non-negative).
`totalTime` Numeric scalar or numeric vector of total operating times.

Value

Numeric scalar: MTTF. If number of failures is zero, returns Inf (with a warning).

Examples

```
mttf(5, 1000)
mttf(c(2,3), c(500,500))
```

ram

Launch the RAM Analysis Tutorial

Description

This function launches an interactive tutorial on Reliability, Availability, and Maintainability (RAM) analysis.

Usage

```
ram()
```

Value

This function does not return a value.

See Also

<https://paulgovan.github.io/ReliaLearnR/>

Examples

```
if (interactive()) {
  ram()
}
```

`rbd`*Launch the Reliability Block Diagram Tutorial*

Description

This function launches an interactive tutorial on Reliability Block Diagrams and system reliability, covering series, parallel, mixed, and k-out-of-n configurations as well as an introduction to Fault Tree Analysis.

Usage

```
rbd()
```

Value

This function does not return a value.

See Also

<https://paulgovan.github.io/ReliaLearnR/>

Examples

```
if (interactive()) {  
  rbd()  
}
```

`rel`*Reliability (1 - outage / total)*

Description

Reliability is the probability that an item will perform its intended function without failure over a specified period under stated conditions. This function computes reliability given outage times and total times.

Usage

```
rel(outageTime, totalTime)
```

Arguments

`outageTime` Numeric scalar or numeric vector of forced outage times.
`totalTime` Numeric scalar or numeric vector of total times (same units as `outageTime`).

Value

Numeric scalar: reliability for the period (between 0 and 1).

Examples

```
rel(100, 1000)
rel(c(10,20), c(500, 600))
```

rs

Launch the Repairable Systems Analysis Tutorial

Description

This function launches an interactive tutorial on Repairable Systems Analysis.

Usage

```
rs()
```

Value

This function does not return a value.

See Also

<https://paulgovan.github.io/ReliaLearnR/>

Examples

```
if (interactive()) {
  rs()
}
```

rt

Launch the Reliability Testing Tutorial

Description

This function launches the Reliability Testing tutorial

Usage

```
rt()
```

Value

This function does not return a value.

See Also

<https://paulgovan.github.io/ReliaLearnR/>

Examples

```
if (interactive()) {  
  rt()  
}
```

Index

avail, 2

fr, 3

lda, 3

mtbf, 4

mttf, 4

ram, 5

rbd, 6

rel, 6

rs, 7

rt, 7