Package 'CARMS'

May 7, 2026

Version 1.0.1

Date 2024-4-4

Title Continuous Time Markov Rate Modeling for Reliability Analysis

Description Emulation of an application originally created by Paul Pukite. Computer Aided Rate Modeling and Simulation. Jan Pukite and Paul Pukite, (1998, ISBN 978-0-7803-3482), William J. Stewart, (1994, ISBN: 0-691-03699-3).

License GPL (>= 3)

URL http://www.openreliability.org

LazyLoad yes

Imports Rcpp, methods, diagram

Author David Silkworth [aut],
Paul Pukite [aut],
Jacob Ormerod [cre],
OpenReliability.org [cph]

Maintainer Jacob Ormerod <jake@openreliability.org>

NeedsCompilation yes

LinkingTo Rcpp, RcppArmadillo

Repository CRAN

Date/Publication 2024-04-04 18:03:03 UTC

Contents

`carms.arrow` *Modify an* `"carms"` *Object adding a transition as an arrow element*

Description

This function defines a transition, by attributes of source (from) and destination (to), rate value, connecting arrow arc, position and diagram label.

Usage

```
carms.arrow(x, from, to, rate, arc=0.35, arrow.position=0.5, label="")
```

Arguments

`x`	An `"carms"` object as initialized by carms.make and modified by previous carms.state, carms.base and/or carms.arrow calls
`from`	The state number (as determined by order of entry) of the source state for transition.
`to`	The state number (as determined by order of entry) of the destination state for transition.
`rate`	A base label or formula (either as a string or as global environment labels) utilizing base labels. Alternatively, a single value unassigned as a base.
`arc`	A convexity/concavity value for the connecting transition arc often determined by trial and modification of diagram development.
`arrow.position`	A value from 0 to 1 for the location of the arrow on a transition arc as well as the transition label often determined by trial and modification of diagram development.
`label`	A string defining the transition often as a rate. The rate and label may often be identical, but they are treated in different ways.

Value

The input object will be returned with the addition this arrow information in the `"carms$arrows$arrow"` list element.

References

Jan Pukite and Paul Pukite (1998), "Modeling for Reliability Analysis", IEEE Press, New York

William J. Stewart (1994), "Introduction to the numerical solution of Markov chains", Princeton University Press, Princeton

Examples

```
SiSimpl<-carms.make(title="Parallel          Identical components", diagram_grid=c(5,8))
SiSimpl<-carms.state(SiSimpl, prob=1, name="P1", size=7, h2w=14/20, position=c(2,3) )
SiSimpl<-carms.state(SiSimpl, prob=0, name="P2", size=7, h2w=14/20, position=c(6,3) )
SiSimpl<-carms.base(SiSimpl, 1, time_units="hours",  description="Failure rate")
SiSimpl<-carms.arrow(SiSimpl, from=1, to=2, rate="B1",label="B1")
```

`carms.base` *Modify an* `"carms"` *Object adding a base element*

Description

This function establishes a base rate that can be called, often in multiples or combination, during transition definition.

Usage

```
carms.base(x, value, time_units=NULL, base_label=NULL, description="")
```

Arguments

`x`	An `"carms"` object as initialized by carms.make and modified only by previous carms.state or carms.base calls
`value`	A rate value
`time_units`	A string establishing the units of time measure. This value is requried on only the first carms.base call on a carms object. It will be ignored in subsequent carms.base calls. The units string provided here will be used for the x-axis label of the simulation plot.
`base_label`	An optional string that can be used as a valid R object name holding this value.
`description`	An optional string providing more information about the base (never seen on diagram nor plot).

Value

The input object will be returned with the addition this base information in the `"carms$base"` list element. Also, any base_label provided will be added to the `"carms$base"` list element as well.

References

Jan Pukite and Paul Pukite (1998), "Modeling for Reliability Analysis", IEEE Press, New York

William J. Stewart (1994), "Introduction to the numerical solution of Markov chains", Princeton University Press, Princeton

Examples

```
SiSimpl<-carms.make(title="Parallel          Identical components", diagram_grid=c(5,8))
SiSimpl<-carms.state(SiSimpl, prob=1, name="P1", size=7, h2w=14/20, position=c(2,3) )
SiSimpl<-carms.state(SiSimpl, prob=0, name="P2", size=7, h2w=14/20, position=c(6,3) )
SiSimpl<-carms.base(SiSimpl, 1, time_units="hours",  description="Failure rate")
```

`carms.make` *Create a* `carms` *Object for continuous time Markov rate modeling*

Description

This function creates an initial carms object

Usage

```
carms.make(title=NULL, diagram_grid=c(11,12),...)
```

Arguments

`title`	A title that will appear on simulation plot(s)
`diagram_grid`	A two element vector defining a grid for placement of state objects (rows, columns)
`...`	A list argument enabling modification of certain defaults (not yet implemented)

Details

This initial carms object is required for the addition of state, base and arrow elements.

Value

A named list of class `"carms"`.

References

Jan Pukite and Paul Pukite (1998), "Modeling for Reliability Analysis", IEEE Press, New York

William J. Stewart (1994), "Introduction to the numerical solution of Markov chains", Princeton University Press, Princeton

Examples

```
jeep<-carms.make("jeep    Tire configurations")

SiSimpl<-carms.make(title="Parallel          Identical components", diagram_grid=c(5,8))
```

carms.state *Modify an "carms" Object adding a state element*

Description

This function defines a state with attributes for diagram name, size, proportion and placement in a diagram.

Usage

```
carms.state(x, prob,  name, size=4, h2w=21/24, position,
Pfunction=NULL, plot.line.color=NULL,  description="")
```

Arguments

x	An "carms" object as initialized by carms.make or modified only by previous carms.state calls
prob	An initial probability.
name	A string that will be used to identify the state on a diagram.
size	A relative value for the state element often determined by trial and modification of diagram development.
h2w	A proportion value for the state element often determined by trial and modification of diagram development.
position	A vector of (column, row) related to the diagram grid and often determined by trial and modification of diagram development.
Pfunction	A string defining a function providing for an alternative probability determination after simulation, typically accumulating the probabilty values of several other states.
plot.line.color	A string providing for an override of the default color palatte designation for this particular state on the simulation plot.
description	A string providing more information about the state (never seen on diagram nor plot).

Value

The input object will be returned with the addition this state information in the "carms$state" list element.

References

Jan Pukite and Paul Pukite (1998), "Modeling for Reliability Analysis", IEEE Press, New York

William J. Stewart (1994), "Introduction to the numerical solution of Markov chains", Princeton University Press, Princeton

Examples

```
SiSimpl<-carms.make(title="Parallel          Identical components", diagram_grid=c(5,8))
SiSimpl<-carms.state(SiSimpl, prob=1, name="P1", size=7, h2w=14/20, position=c(2,3) )
SiSimpl<-carms.state(SiSimpl, prob=0, name="P2", size=7, h2w=14/20, position=c(6,3) )
```

`diagram` *Deliver a Markov diagram display of a carms object.*

Description

This function is simply an alias to diagram.carms since diagram is not avaliable as an S3 method.

Usage

```
diagram(x, text.size=0.7, rate.text.y.shift=0.7,shadow=FALSE )
```

Arguments

`x`	A `"carms"` object as initialized by carms.make and modified by previous carms.state and/or carms.base and carms.arrow calls
`text.size`	A relative value controlling the size of text displayed on the diagram.
`rate.text.y.shift`	A relative value controlling the separation of arrow label text from the arrow itself.
`shadow`	A logical determining whether each state on the diagram is depicted with a shadow.

Value

This function returns no value. The image on the generated graphics device is typically used for progressive editing of state size, h2w, and position arguments as well as arrow arc, and arrow.position arguments.

References

Jan Pukite and Paul Pukite (1998), "Modeling for Reliability Analysis", IEEE Press, New York

William J. Stewart (1994), "Introduction to the numerical solution of Markov chains", Princeton University Press, Princeton

Examples

```
SiSimpl<-carms.make(title="Parallel          Identical components", diagram_grid=c(5,8))
SiSimpl<-carms.state(SiSimpl, prob=1, name="P1", size=7, h2w=14/20, position=c(2,3) )
SiSimpl<-carms.state(SiSimpl, prob=0, name="P2", size=7, h2w=14/20, position=c(6,3) )
SiSimpl<-carms.base(SiSimpl, 1, time_units="hours",  description="Failure rate")
SiSimpl<-carms.arrow(SiSimpl, from=1, to=2, rate="B1",label="B1")

diagram(SiSimpl)
```

`diagram.carms` *Deliver a Markov diagram display of a carms object.*

Description

This function delivers, at times with a preceeding dev.new call, a diagram (often in development) to an R graphics device.

Usage

```
diagram.carms(x, text.size=0.7, rate.text.y.shift=0.7,shadow=FALSE )
```

Arguments

`x`	A `"carms"` object as initialized by carms.make and modified by previous carms.state and/or carms.base and carms.arrow calls
`text.size`	A relative value controlling the size of text displayed on the diagram.
`rate.text.y.shift`	A relative value controlling the separation of arrow label text from the arrow itself.
`shadow`	A logical determining whether each state on the diagram is depicted with a shadow.

Value

This function returns no value. The image on the generated graphics device is typically used for progressive editing of state size, h2w, and position arguments as well as arrow arc, and arrow.position arguments.

References

Jan Pukite and Paul Pukite (1998), "Modeling for Reliability Analysis", IEEE Press, New York

William J. Stewart (1994), "Introduction to the numerical solution of Markov chains", Princeton University Press, Princeton

Examples

```
SiSimpl<-carms.make(title="Parallel          Identical components", diagram_grid=c(5,8))
SiSimpl<-carms.state(SiSimpl, prob=1, name="P1", size=7, h2w=14/20, position=c(2,3) )
SiSimpl<-carms.state(SiSimpl, prob=0, name="P2", size=7, h2w=14/20, position=c(6,3) )
SiSimpl<-carms.base(SiSimpl, 1, time_units="hours",  description="Failure rate")
SiSimpl<-carms.arrow(SiSimpl, from=1, to=2, rate="B1",label="B1")

diagram.carms(SiSimpl, shadow=TRUE)
```

`examples` *Handle the installed examples in a carms project.*

Description

This function delivers, a list of the examples, will "run" (source) an example script, copy an installed example to the working directory, or display the examples directory location.

Usage

```
examples(action="list", file=NULL, package="CARMS")
```

Arguments

`action`	A string indicating how the function is to operate. "list", "run", "copy" and "dir" are expected arguments.
`file`	An integer representing the file row in the "list" dataframe, or string providing the exact file name desired to be sourced.
`package`	A string naming the package.

Details

This function is provided as a crutch for users of the CARMS package who may have no prior experience with R. To them a sourced script has been "run". It is also desirable for them to copy example scripts into their working directory for examination and edit, rather than accessing the examples (and potentially modifying them) from the installed examples directory.

Value

For the "list" action a dataframe with single column named "examples". For the "dir" action a string providing the examples directory reference. For a successful "run" action a 1 will be returned. Of course the diagram and plot will also display. For a successful "copy" action TRUE will be returned.

References

Jan Pukite and Paul Pukite (1998), "Modeling for Reliability Analysis", IEEE Press, New York

William J. Stewart (1994), "Introduction to the numerical solution of Markov chains", Princeton University Press, Princeton

Examples

```
examples() # will return a dataframe listing the available examples
examples("dir") # will return the installed examples directory reference.

examples("run", 1) # will source the first example
```

---

examples.carms *Handle the installed examples in a carms project.*

Description

This function delivers, a list of the examples, will "run" (source) an example script, copy an installed example to the working directory, or display the examples directory location.

Usage

```
examples.carms(action="list", file=NULL, package="CARMS")
```

Arguments

action	A string indicating how the function is to operate. "list", "run", and "dir" are expected arguments.
file	An integer representing the file row in the "list" dataframe, or string providing the exact file name desired to be sourced.
package	A string naming the package.

Details

This function is provided as a crutch for users of the CARMS package who may have no prior experience with R. To them a sourced script has been "run". It is also desirable for them to copy example scripts into something like their working directory for examination and edit. The "dir" output provides information for the user to locate the installed examples. It is left to the user as to what to do with the example scripts.

Value

For the "list" action a dataframe with single column named "examples". For the "dir" action a string providing the examples directory reference. For a successful "run" action a 1 will be returned. Of course the diagram and plot will also display.

References

Jan Pukite and Paul Pukite (1998), "Modeling for Reliability Analysis", IEEE Press, New York

William J. Stewart (1994), "Introduction to the numerical solution of Markov chains", Princeton University Press, Princeton

Examples

```
examples.carms() # will return a dataframe listing the available examples
examples.carms("dir") # will return the installed examples directory reference.

examples.carms("run", 1) # will source the first example
```

`matP` *Extract the probability matrix from a completed carms object with a simulation.*

Description

This function extracts the probability matrix from a completed carms object with a simulation.

Usage

```
matP(x)
```

Arguments

`x`	A completed carms object with a simulation.

Value

A matrix of probabilities for each state at each time step of the simulation.

References

Jan Pukite and Paul Pukite (1998), "Modeling for Reliability Analysis", IEEE Press, New York

William J. Stewart (1994), "Introduction to the numerical solution of Markov chains", Princeton University Press, Princeton

Examples

```
SiSimpl<-carms.make(title="Parallel          Identical components", diagram_grid=c(5,8))
SiSimpl<-carms.state(SiSimpl, prob=1, name="P1", size=7, h2w=14/20, position=c(2,3) )
SiSimpl<-carms.state(SiSimpl, prob=0, name="P2", size=7, h2w=14/20, position=c(6,3) )
SiSimpl<-carms.base(SiSimpl, 1, time_units="hours",  description="Failure rate")
SiSimpl<-carms.arrow(SiSimpl, from=1, to=2, rate="B1",label="B1")
SiSimpl<-simulate(SiSimpl, solution="rk", mission_time=200)
matP(SiSimpl)
```

plot.carms *Deliver a plot of the simulation of a carms object.*

Description

This function delivers a plot of the varying probability of each state over the simulated mission history.

Usage

```
## S3 method for class 'carms'
plot(x, spline_curve=TRUE, knots=10, ...)
```

Arguments

x	A completed "carms" object including a simulation.
spline_curve	A logical determining whether to generate a smoothed spline curve from the simulation data (alternatively line segments from step to step will be displayed).
knots	An integer value of a fraction of simulation intervals to use for determination of a smoothed spline.
...	A list argument enabling modification of certain defaults (not yet implemented)

Details

This function is specified as an S3 method so can be called as simply plot().

Value

This function returns a matrix of probabilities for each state at each time step to the carms$simulation list element.

References

Jan Pukite and Paul Pukite (1998), "Modeling for Reliability Analysis", IEEE Press, New York

William J. Stewart (1994), "Introduction to the numerical solution of Markov chains", Princeton University Press, Princeton

Examples

```
SiSimpl<-carms.make(title="Parallel          Identical components", diagram_grid=c(5,8))
SiSimpl<-carms.state(SiSimpl, prob=1, name="P1", size=7, h2w=14/20, position=c(2,3) )
SiSimpl<-carms.state(SiSimpl, prob=0, name="P2", size=7, h2w=14/20, position=c(6,3) )
SiSimpl<-carms.base(SiSimpl, 1, time_units="hours",  description="Failure rate")
SiSimpl<-carms.arrow(SiSimpl, from=1, to=2, rate="B1",label="B1")
SiSimpl<-simulate(SiSimpl, solution="rk", mission_time=200)
```

```
plot.carms(SiSimpl)
plot(SiSimpl)
```

`simulate` *Generate a simulation of a carms object.*

Description

This function calls for the simulation on a completed carms object.

Usage

```
simulate(x, solution, mission_time, intervals=50, cycles=2000)
```

Arguments

`x`	A completed `"carms"` object as initialized by carms.make and modified by previous carms.state, carms.base and carms.arrow calls
`solution`	A string of "rk", "bd", or "chain" indicating the method of obtaining the simulation solution.
`mission_time`	A time value for the extent of mission history for the simulation
`intervals`	An integer value for the number of intervals over the mission history to calculate the simulation.
`cycles`	An integer value (usually in the thousands) impacting the number of simulations run only when using the chain solution.

Details

It was chosen not to register this functionThis function, due to differences in fundimental arguments with stats::simulate. For this reason upon loading the CARMS library one will notice that function simulate is masked from the stats library.

Value

This function returns a matrix of probabilities for each state at each time step to the carms$simulation list element.

References

Jan Pukite and Paul Pukite (1998), "Modeling for Reliability Analysis", IEEE Press, New York

William J. Stewart (1994), "Introduction to the numerical solution of Markov chains", Princeton University Press, Princeton

Examples

```
SiSimpl<-carms.make(title="Parallel          Identical components", diagram_grid=c(5,8))
SiSimpl<-carms.state(SiSimpl, prob=1, name="P1", size=7, h2w=14/20, position=c(2,3) )
SiSimpl<-carms.state(SiSimpl, prob=0, name="P2", size=7, h2w=14/20, position=c(6,3) )
SiSimpl<-carms.base(SiSimpl, 1, time_units="hours",  description="Failure rate")
SiSimpl<-carms.arrow(SiSimpl, from=1, to=2, rate="B1",label="B1")
SiSimpl<-simulate(SiSimpl, solution="rk", mission_time=200)
```

`simulate.carms` *Generate a simulation of a carms object.*

Description

This function calls for the simulation on a completed carms object.

Usage

```
simulate.carms(x, solution, mission_time, intervals=50, cycles=2000)
```

Arguments

`x`	A completed `"carms"` object as initialized by carms.make and modified by previous carms.state, carms.base and carms.arrow calls
`solution`	A string of "rk", "bd", or "chain" indicating the method of obtaining the simulation solution. Although not intended for production work, it is possible to enter a string of "chain_R" to execute the formative R code for the stochastic chain solution.
`mission_time`	A time value for the extent of mission history for the simulation
`intervals`	An integer value for the number of intervals over the mission history to calculate the simulation.
`cycles`	An integer value (usually in the thousands) impacting the number of simulations run only when using the chain solution.

Details

It was chosen not to register this functionThis function, due to differences in fundimental arguments with stats::simulate.

Value

This function returns a matrix of probabilities for each state at each time step to the carms$simulation list element.

References

Jan Pukite and Paul Pukite (1998), "Modeling for Reliability Analysis", IEEE Press, New York

William J. Stewart (1994), "Introduction to the numerical solution of Markov chains", Princeton University Press, Princeton

Examples

```
SiSimpl<-carms.make(title="Parallel          Identical components", diagram_grid=c(5,8))
SiSimpl<-carms.state(SiSimpl, prob=1, name="P1", size=7, h2w=14/20, position=c(2,3) )
SiSimpl<-carms.state(SiSimpl, prob=0, name="P2", size=7, h2w=14/20, position=c(6,3) )
SiSimpl<-carms.base(SiSimpl, 1, time_units="hours",  description="Failure rate")
SiSimpl<-carms.arrow(SiSimpl, from=1, to=2, rate="B1",label="B1")
SiSimpl<-simulate.carms(SiSimpl, solution="rk", mission_time=200)
```

Index