Package 'dmutate'

April 23, 2021

Type Package

Title Mutate Data Frames with Random Variates

Version 0.1.3

Imports dplyr (>= 0.7.4), MASS

Depends methods

Suggests testthat

Maintainer Kyle T Baron <kylebtwin@imap.cc>

Description Work within the 'dplyr' workflow to add random variates to your data frame. Variates can be added at any level of an existing column. Also, bounds can be specified for simulated variates.

URL https://github.com/kylebaron/dmutate

BugReports https://github.com/kylebaron/dmutate/issues

License GPL (>= 2)

RoxygenNote 7.1.1

Encoding UTF-8

NeedsCompilation no

Author Kyle T Baron [aut, cre, cph]

Repository CRAN

Date/Publication 2021-04-22 22:20:03 UTC

R topics documented:

as_idata *Create individual data frame from a covset object*

Description

Create individual data frame from a covset object

Usage

```
as_idata(.covset, .n)
```

Arguments

.covset	a covset object
.n	number of IDs to simulate

Examples

```
cov1 <- covset(Y ~ rbinomial(0.2), Z ~ rnorm(2,2))

as_idata(cov1, 10)
```

build_covform *Build a object or formula to use with covset.*

Description

build_covform formulates then parses a formula that can be used in a covset. build_covobj just assembles the object directly.

Usage

```
build_covform(
  var,
  dist,
  args,
  lower = NULL,
  upper = NULL,
  by = NULL,
  envir = parent.frame()
)

build_covobj(
  var,
  dist,
```

```
  args,
  upper = NULL,
  lower = NULL,
  by = NULL,
  envir = parent.frame()
)
```

Arguments

var	variable name, character
dist	distribution function name
args	character vector of arguments for dist
lower	lower limits for var
upper	upper limits for var
by	grouping variable
envir	environment for resolving symbols in expressions

Details

When length of var is greater than one, both lower and upper must be named vectors when specifiation is made. However, it is acceptable to specify nothing or to use unnamed limits when the lenght of var is 1.

Examples

```
build_covform("WT", "rnorm", c("mu = 80", "sd = 40"), lower = 40, upper = 140)
build_covform("WT", "rnorm", "80,40", lower = 40, upper = 140)

build_covobj("WT", "rnorm", "80,40", lower = 40, upper = 140)
```

---

covset *Covobj and covset objects.*

Description

Covobj and covset objects.

Create a set of covariates.

Usage

```
new_covobj(x, envir = parent.frame(), ...)

## S3 method for class 'covobj'
print(x, ...)

## S4 method for signature 'covobj'
as.list(x, ...)

## S4 method for signature 'covset'
as.list(x, ...)

## S3 method for class 'covset'
print(x, ...)

covset(..., envir = parent.frame())

rvset(...)

as.covset(x)
```

Arguments

x	a formula; may be quoted
envir	for formulae
...	formulae to use for the covset

Details

rvset is an alias for covset.

Examples

```
obj <- new_covobj(Y[0,80] ~ rnorm(20,50))

obj

as.list(obj)

a <- Y ~ runif(0,1)
b <- Z ~ rbeta(1,1)

set <- covset(a,b)

set

as.list(set)
```

dmutate *mutate a data frame, adding random variables.*

Description

mutate a data frame, adding random variables.

Apply formulae to a data frame

Usage

```
dmutate(data, ...)
```

Arguments

data	a data frame
...	formulae and other arguments for mutate_random

Examples

```
idata <- dplyr::data_frame(ID = 1:10)

dmutate(idata, y ~ rbinomial(0.5), wt ~ rnorm(mu,sd),
        envir = list(mu = 50, sd = 20))
```

mutate_random *Add random variates to a data frame.*

Description

Add random variates to a data frame.

Usage

```
mutate_random(data, input, ...)

## S4 method for signature 'data.frame,formula'
mutate_random(data, input, ...)

## S4 method for signature 'data.frame,character'
mutate_random(data, input, envir = parent.frame(), ...)

## S4 method for signature 'data.frame,list'
mutate_random(data, input, ...)
```

```
## S4 method for signature 'data.frame,covset'
mutate_random(data, input, ...)

## S4 method for signature 'data.frame,covobj'
mutate_random(data, input, envir = parent.frame(), ...)
```

Arguments

data	the data.frame to mutate
input	an unquoted R formula; see details
...	additional inputs
envir	environment for object lookup

Examples

```
data <- data.frame(ID=1:10, GROUP = sample(c(1,2,3),10,replace=TRUE))

mutate_random(data, AGE[40,90] ~ rnorm(55,50))
mutate_random(data, RE ~ rbeta(1,1) | GROUP)

e <- list(lower=40,upper=140,mu=100,sd=100)

egfr <- covset(EGFR[lower,upper] ~ rnorm(mu,sd))

mutate_random(data,egfr,envir=e)
```

---

rbinomial *Simulate from binomial distribution.*

Description

Wrapper for rbinom with trial size of 1.

Usage

```
rbinomial(n, p, ...)

rbern(n, p, ...)
```

Arguments

n	number of variates
p	probability of success
...	passed along as appropriate

Details

The `size` of each trial is always 1.

`rmvnorm` *Simulate from multivariate normal distribution.*

Description

Simulate from multivariate normal distribution.

Usage

```
rmvnorm(n, mu, Sigma)

rlmvnorm(n, ...)

rmassnorm(n, ...)

rlmassnorm(n, ...)
```

Arguments

`n`	number of variates
`mu`	vector of means
`Sigma`	variance-covariance matrix with number of columns equal to length of `mu`
`...`	arguments passed to `rmvnorm`

Details

`rlmvnorm` is a multivariate log normal.

`rmassnorm` and `rlmassnorm` simulate the multivariate normal using the `MASS` package.

Value

Returns a matrix of variates with number of rows equal to `n` and mumber of columns equal to length of `mu`.

Index