Package 'pmxcv'

August 25, 2025

Title Integration-Based Coefficients of Variation

Version 0.0.2

Description Estimate coefficient of variation percent (CV%) for any arbitrary distribution, including some built-in estimates for commonly-used transformations in pharmacometrics. Methods are described in various sources, but applied here as summarized in: Prybylski, (2024) <doi:10.1007/s40262-023-01343-2>.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.2

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author John Prybylski [aut, cre, cph] (ORCID: <https://orcid.org/0000-0001-5802-0539>)

Maintainer John Prybylski <jprybylski@gmail.com>

Repository CRAN

Date/Publication 2025-08-25 13:50:08 UTC

Contents

dist.intcv *Built-in integration-based %CV functions*

Description

Built-in integration-based %CV functions

Usage

```
dist.intcv(
  dist = "log",
  ...,
  exact = ifelse(dist == "log", TRUE, FALSE),
  lambda = NULL,
  fun = FALSE
)
```

Arguments

dist	Selection of built-in distributions.
...	passed to moment()
exact	If there is an exact moment generating function, use that. Default TRUE only for log
lambda	shape parameter for nonmemboxcox()
fun	return function (for use in invcv())

Value

Percent CV

Examples

```
dist.intcv("log", v = 0.2)
dist.intcv("logit", u = 0.5, v = 0.3)
```

dist.moment *Built-in moment functions*

Description

Built-in moment functions

Usage

```
dist.moment(
  dist = "log",
  ...,
  exact = ifelse(dist == "log", TRUE, FALSE),
  lambda = NULL
)
```

Arguments

dist	Selection of built-in distributions.
...	passed to moment()
exact	If there is an exact moment generating function, use that. Default TRUE only for log
lambda	shape parameter for nonmemboxcox()

Value

moment

Examples

```
dist.moment("log", u = 2, v = 0.2, n = 2)
dist.moment("logit", u = 0.5, v = 0.2, n = 1)
```

---

intcv *Integration-based CV%*

Description

Integration-based CV%

Usage

```
intcv(...)
```

Arguments

...	Arguments passed to moment()

Value

Percent CV

Examples

```
intcv(u = 1, v = 0.2, pdist = exp, qdist = log)
```

invcv *Variance from CV%*

Description

Variance from CV%

Usage

```
invcv(cvfun, cv, verbose = FALSE, ...)
```

Arguments

cvfun	intcv()-based function
cv	CV% generated from cvfun
verbose	extra output
...	Other parameters to pass to cvfun

Value

Best-fit variance

Examples

```
logcv <- dist.intcv("log", fun = TRUE)
invcv(logcv, cv = 30)
```

moment *Moment function*

Description

Moment function

Usage

```
moment(...)
```

Arguments

...	all arguments passed to moment_f()

Value

moment

Examples

```
moment(n = 3, u = 1, v = 0.2, pdist = exp, qdist = log)
```

`moment_f` *Integratable moment function*

Description

Integratable moment function

Usage

```
moment_f(x, u, v, n, pdist, qdist)
```

Arguments

x	numeric vector
u	mean
v	variance
n	moment number
pdist	un-transform function for transformed random variable (eg, exp())
qdist	transform function (eg, log())

Value

Point result of the moment function

Examples

```
moment_f(0, u = 1, v = 0.2, n = 1, pdist = exp, qdist = log)
```

`nonmemboxcox` *Box-Cox transform typically used in NONMEM*

Description

Parameters are typically treated as lognormally-distributed by NONMEM users. Box-Cox transforms are typically applied to the exponentiated individual ETA parameters; this means the parameter is neither Box-Cox distributed nor lognormally-distributed, but both. To get the "Box-Cox Transform" as it would be relevant for CV% calculation, these properties have to be considered.

Usage

```
nonmemboxcox(x, lambda, theta = 1, inv = FALSE)
```

Arguments

x	random vector. Must be positive.
lambda	shape parameter
theta	centrality parameter
inv	inverse transform

Value

Box-Cox transformed or untransformed vector

Examples

```
y <- nonmemboxcox(1.5, lambda = 0.5, theta = 1)
nonmemboxcox(y, lambda = 0.5, theta = 1, inv = TRUE)
```

---

numcv *Numeric CV% of a sample*

Description

Numeric CV% of a sample

Usage

```
numcv(x, ...)
```

Arguments

x	numeric vector
...	other arguments for sd() and mean()

Value

Percent cv

Examples

```
test_x <- rnorm(1000, mean=50, sd=5)
cv <- numcv(test_x)
cv # expect ~ 10(%)
```

Index