Package ‘nisone’

September 8, 2026

Title Inference for Samples of Size at Least One

Version 0.0.2

Description Provides different interval estimates of a location parameter when the sample size is one or more. These include classical methods when n=1, new Bayesian analogues of these classical methods, and extensions of these methods to larger sample sizes in the normal case. Other functions calculate Bayes factors based on t-statistics, and implement the (generalized) inverse normal distribution. These methods are described in Gerard (2026) <doi:10.48550/arXiv.2607.25007>.

License GPL (>= 3)

Encoding UTF-8

URL https://dcgerard.github.io/nisone/

BugReports https://github.com/dcgerard/nisone/issues

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

Depends R (>= 3.5)

Config/roxygen2/version 8.1.0

VignetteBuilder knitr

NeedsCompilation no

Author David Gerard [aut, cre] (ORCID: <https://orcid.org/0000-0001-9450-5023>)

Maintainer David Gerard <gerard.1787@gmail.com>

Repository CRAN

Date/Publication 2026-09-08 12:40:02 UTC

Contents

`aug_t` *Augmented t-interval*

Description

Calculates a t-interval using augmented data `c(x,A)`. The multiplier of this interval bounds the level above `level`, so these intervals are typically conservative. This method is very fast for `n <= 100` and `level %in% c(0.8, 0.9, 0.95, 0.99)` because I saved those multipliers in an internal dataset. But it can be slow (on the order of a second) for other confidence levels or larger sample sizes.

Usage

```
aug_t(x, A = 0, level = 0.95, wt = 1)
```

Arguments

`x`	The vector of data
`A`	The prior mean
`level`	The level of the interval
`wt`	Weight for A. Don't touch this unless you know what you are doing.

Details

Suppose $X_1, \ldots, X_n \sim N(\mu, \sigma^2)$. Suppose we have prior value A. This provides intervals of the form

$$\hat{\mu} \pm \eta \hat{\sigma} / \sqrt{n+1},$$

where $\hat{\mu}$ and $\hat{\sigma}$ are the sample mean and sample standard deviation of the augmented data $X_1, \ldots, X_n, A$, and η is chosen large enough to maintain the confidence level at all parameter values.

The `wt` argument allows for more copies of A to be included in the data augmentation. But it doesn't work well with more data augmentation so you should not set it above 1. Though, you can set `wt` to be between 0 and 1 (to have less data augmentation) and this does seem to work pretty well, but I haven't studied it extensively, so use at your own risk.

Note that you have to choose A *before* seeing the data. If you choose it based on the data, then you no longer maintain the confidence level.

Value

The augmented t-interval of the specified level.

Author(s)

David Gerard

Examples

```
set.seed(1)
## A is exactly correct
x <- rnorm(10)
aug_t(x)
stats::t.test(x)$conf.int

## A is very wrong
x <- rnorm(10)
aug_t(x, A = 100)
stats::t.test(x)$conf.int
```

`bci1` *Bayesian credible interval when n=1*

Description

Suppose $X \sim N(\mu, \sigma^2)$. This gives you credible intervals of a specified level when we use prior $\sigma = |\mu|$ with probability 1, and $\pi(\mu) = |\mu - A|^{-1}$. See the `n1post()` for access to the full posterior distribution.

Usage

```
bci1(x, A = 0, level = 0.95)
```

Arguments

`x`	a double
`A`	The prior center.
`level`	The level of the credible interval

Value

The credible interval of the specified level.

Author(s)

David Gerard

See Also

`n1post()`: Functions to access the full posterior distribution.

Examples

```
bci1(x = 1, A = 0)
```

`bcin` *Bayesian credible interval when n >= 1*

Description

Let $X_i \sim N(\mu, \sigma^2)$. Let $Y_i = X_i - A$, $\beta = \mu - A$, and $\nu = |\mu - A|/\sigma$. Then, for a fixed value of ν, this comes up with a $(1-\alpha)100\%$ posterior credible interval of μ where the prior is $\pi(\beta) = |\beta|^{-1}$ or $\pi(\mu) = |\mu - A|^{-1}$. The full posterior distribution is generalized inverse normal (implemented by `ginvnorm`) with shape parameter $n+1$, inverse mean of `sum(y) / sum(y^2)`, and inverse variance `1 / (nu^2 * sum(y^2))`.

Usage

```
bcin(x, A = 0, nu = 1, level = 0.95)
```

Arguments

`x`	a double
`A`	The prior center.
`nu`	The fixed value of nu.
`level`	The level of the credible interval

Value

The credible interval of the provided level.

Author(s)

David Gerard

See Also

`ginvnorm()`: The generalized inverse normal distribution.

Examples

```
set.seed(1)
x <- stats::rnorm(4, mean = 1, sd = 1)
bcin(x = x, A = 3)
```

---

bft *Bayes factor from t-statistic*

Description

Given a user-provided prior on the standardized effect size, will calculate the Bayes factor.

Usage

```
bft(t, nu, nd, prior = NULL)
```

Arguments

`t`	The observed t-statistic.
`nu`	The degrees of freedom of the t-statistic. This should be $n-1$ in the one-sample case. This should be $n_1 + n_2 - 2$ in the two-sample case.
`nd`	Effective sample size. This should be n in the one-sample case. This should be $(\frac{1}{n_1} + \frac{1}{n_2})^{-1}$ in the two-sample case.
`prior`	The prior on the standardized effect size, δ. In the one-sample case this is the prior over $\delta = \frac{\mu - \mu_0}{\sigma}$, where μ_0 is the null value. In the two-sample case this is the prior over $\delta = \frac{\mu_1 - \mu_2}{\sigma}$, where μ_1 and μ_2 are the means of the two-samples. The default prior is a Cauchy, but you can put in any density function you want. If your prior has pointmasses in it, this function won't work.

Details

Let δ be the standardized effect size, let σ^2 be the variance (assumed equal in two-sample case). In the one-sample case, $\delta = \frac{\mu - \mu_0}{\sigma}$, where μ_0 is the null value. In the two-sample case, $\delta = \frac{\mu_1 - \mu_2}{\sigma}$, where μ_1 and μ_2 are the means of the two-samples. We place the prior $1/\sigma^2$ under the null and $\pi(\delta)/\sigma^2$ under the alternative, for some arbitrary density $\pi(\cdot)$. Given this setting, the Bayes factor is a function of the t-statistic. We calculate it via numeric integration.

Value

The Bayes factor to a corresponding t-statistic.

Author(s)

David Gerard

References

- Gronau, Q. F., Ly, A., & Wagenmakers, E. J. (2020). Informed Bayesian t-tests. *The American Statistician*. doi:10.1080/00031305.2018.1562983

Examples

```
# One sample t, n = 10, t-statistic = 2
bft(t = 2, nu = 10 - 1, nd = 10)

# Two sample t, n1 = 10, n2 = 8, t-statistic = 2
bft(t = 2, nu = 10 + 8 - 2, nd = 1 / (1 / 10 + 1 / 8))
```

---

blevel — *True level of the Bayesian credible interval at a given coefficient of variation*

Description

True level of the Bayesian credible interval at a given coefficient of variation

Usage

```
blevel(level, nu)
```

Arguments

`level`	The credible interval level.
`nu`	(mu - A)/sigma

Value

The true level of the (1-alpha) credible interval.

Author(s)

David Gerard

Examples

```
blevel(0.95, 0.5)
blevel(0.95, 1)
blevel(0.95, 1.5)
```

ci1 *n=1 confidence interval*

Description

When you have one observation, X from some symmetric distribution with center μ, for a pre-specified A, produces confidence intervals for the center form $X \pm \eta|X - A|$ (`type = "x"`) or $(X + A)/2 \pm \eta|X - A|$ (`type = "ave"`). The average intervals have smaller width on average, and so are the default. We allow X to either come from a normal distribution, a Cauchy, or a uniform. Note that if you use `family = "uniform"` then these intervals are valid confidence intervals for the median/mode for *any* symmetric unimodal density, which I think is really amazing.

Usage

```
ci1(
  x,
  A = 0,
  type = c("ave", "x"),
  family = c("normal", "cauchy", "uniform"),
  level = 0.95
)
```

Arguments

`x`	A vector of single observations.
`A`	Your "prior knowledge" or "augmented data value".
`type`	Either centered at x or the average of x and A
`family`	Either the normal distribution, Cauchy, or uniform.
`level`	The level of the confidence interval.

Value

A matrix of with 4 columns: the data, the center, the lower bound, the upper bound.

Author(s)

David Gerard

References

- Blachman, N., & Machol, R. (1987). Confidence intervals based on one or more observations. *IEEE Transactions on Information Theory*, 33(3), 373-382. doi:10.1109/TIT.1987.1057306

Examples

```
ci1(c(1, 2, 10), type = "x")
ci1(c(1, 2, 10), type = "ave")
```

d_wc *Confidence density based on n=1 confidence interval*

Description

Let $X \sim N(\mu, \sigma^2)$. Consider intervals of the form $X \pm \eta|X|$ or $X/2 \pm \eta|X|$. These intervals define a confidence distribution of μ. This is the density of the confidence distribution of $(\mu - X)/|X|$ or $(\mu - X/2)/|X|$. Please note that μ is random here not X. Also note that this confidence distribution does not exist between the 25th and 75th percentiles.

Usage

```
d_wc(x, center = c("X", "ave"))
```

Arguments

x	The value at which to calculate the density. Only defined for `abs(x) >= 1` when `center = "X"`, and for `abs(x) >= 0.5` when `center = "ave"`.
center	What is the center of the interval? Either `"X"` or `"X/2"` (`"ave"`).

Value

n=1 confidence density.

Author(s)

David Gerard

See Also

`p_wc()`

Examples

```
xseq <- seq(-10, 10, length.out = 500)
dseq <- d_wc(xseq)
plot(xseq, dseq, type = "l")

dseq <- d_wc(xseq, center = "ave")
plot(xseq, dseq, type = "l")
```

`ginvnorm` *The generalized inverse normal distribution*

Description

Density, distribution function, quantile function, and random generation for the *generalized* inverse normal distribution when parameterized by the shape, mean, and standard deviation of the inverse (reciprocal).

Usage

```
dginvnorm(x, alpha, mu = 0, tau = 1, log = FALSE)

pginvnorm(q, alpha, mu = 0, tau = 1, lower.tail = TRUE, subdivisions = 500L)

qginvnorm(p, alpha, mu = 0, tau = 1)

rginvnorm(n, alpha, mu = 0, tau = 1)

mginvnorm(alpha, mu = 0, tau = 1)

xginvnorm(alpha, mu = 0, tau = 1)
```

Arguments

`x, q`	vector of quantiles
`alpha`	vector of shape parameters
`mu`	vector of means of inverse.
`tau`	vector of standard deviations of inverse.
`log`	logical; if `TRUE`, probabilities p are given as log(p).
`lower.tail`	logical; if `TRUE` (default), probabilities are P(X<=x) otherwise, P(X>x).
`subdivisions`	The maximum number of subintervals used in `integrate()`.
`p`	vector of probabilities
`n`	sample size

Value

Either a random sample (`rginvnorm`), the density (`dginvnorm`), the tail probability (`pginvnorm`), or the quantile (`qginvnorm`) of the inverse normal distribution.

Functions

- `dginvnorm()`: Density function.
- `pginvnorm()`: Probability function.
- `qginvnorm()`: Quantile function.
- `rginvnorm()`: Random generation.
- `mginvnorm()`: Mean.
- `xginvnorm()`: Modes.

Author(s)

David Gerard

References

- Robert, C. (1991). Generalized inverse normal distributions. Statistics & Probability Letters, 11(1), 37-41. doi:10.1016/01677152(91)90174P

Examples

```
set.seed(50)
samp <- rginvnorm(n = 1000, alpha = 4, mu = 0.5, tau = 1)
x <- seq(min(samp), max(samp), length.out = 500)
y <- dginvnorm(x = x, alpha = 4, mu = 0.5, tau = 1)
modes <- xginvnorm(alpha = 4, mu = 0.5, tau = 1)
graphics::hist(
  samp,
  freq = FALSE,
  breaks = 100,
  xlab = "x",
  main = "Generalized Inverse Normal Density")
graphics::lines(x, y, col = "#E69F00")
graphics::abline(v = modes, col = "#56B4E9", lty = 2)
```

`idist` *General inverse distribution*

Description

Let X come from some location-scale family

$$f(x) = \frac{1}{\sigma}\rho\left(\frac{X-\mu}{\sigma}\right),$$

for some standard distribution $\rho()$. You provide $\rho()$, μ and σ and these functions will provide the density, distribution, quantile, and random generation for $Z = 1/X$. Convenience arguments for the normal, Cauchy, and uniform are provided.

Usage

```
didist(
  x,
  center = 0,
  scale = 1,
  fam = c("normal", "cauchy", "uniform"),
  ddist = NULL,
  log = FALSE,
  ...
)

pidist(
  q,
  center = 0,
  scale = 1,
  fam = c("normal", "cauchy", "uniform"),
  pdist = NULL,
  lower.tail = TRUE,
  log.p = FALSE,
  ...
)

qidist(
  p,
  center = 0,
  scale = 1,
  fam = c("normal", "cauchy", "uniform"),
  qdist = NULL,
  pdist = NULL,
  ...
)

ridist(
  n,
  center = 0,
  scale = 1,
  fam = c("normal", "cauchy", "uniform"),
  rdist = NULL,
  ...
)
```

Arguments

x, q	vector of quantiles
center	The center parameter μ.
scale	The scale parameter σ.
fam	One of "normal", "cauchy", or "uniform". If ddist, pdist, qdist, or rdist are specified then this argument is ignored.

ddist	Density function of standard distribution $\rho()$. Should have a `log` argument.
log, log.p	logical; if `TRUE`, probabilities p are given as log(p).
...	Additional arguments for `ddist`, `pdist`, `qdist`, and `rdist`.
pdist	Cumulative distribution function of standard distribution.
lower.tail	logical; if `TRUE` (default), probabilities are P(X<=x) otherwise, P(X>x).
p	vector of probabilities
qdist	Quantile function of standard distribution.
n	sample size
rdist	Random generation of standard distribution.

Value

Either a random sample (`ridist`), the density (`didist`), the tail probability (`pidist`), or the quantile (`qidist`) of the inverse distribution.

Functions

- `didist()`: Density function.
- `pidist()`: Distribution function.
- `qidist()`: Quantile function.
- `ridist()`: Random generation.

Author(s)

David Gerard

Examples

```
didist(1, fam = "normal")
didist(1, center = 3, scale = 2, ddist = dt, df = 1)

pidist(1, fam = "normal")
pidist(1, center = 3, scale = 2, pdist = pt, df = 1)

qidist(0.1, fam = "normal")
qidist(0.1, center = 3, scale = 2, qdist = qt, pdist = pt, df = 1)
```

`invnorm` *The inverse normal distribution*

Description

Density, distribution function, quantile function, and random generation for the inverse normal distribution when parameterized by the mean and standard deviation of the inverse (reciprocal).

Usage

```
dinvnorm(x, imean = 0, isd = 1, log = FALSE)

pinvnorm(q, imean = 0, isd = 1, lower.tail = TRUE, log.p = FALSE)

qinvnorm(p, imean = 0, isd = 1)

rinvnorm(n, imean = 0, isd = 1)
```

Arguments

`x`, `q`	vector of quantiles
`imean`	vector of means of inverse.
`isd`	vector of standard deviations of inverse.
`log`, `log.p`	logical; if `TRUE`, probabilities p are given as log(p).
`lower.tail`	logical; if `TRUE` (default), probabilities are P(X<=x) otherwise, P(X>x).
`p`	vector of probabilities
`n`	sample size

Value

Either a random sample (`rinvnorm`), the density (`dinvnorm`), the tail probability (`pinvnorm`), or the quantile (`qinvnorm`) of the inverse normal distribution.

Functions

- `dinvnorm()`: Density function.
- `pinvnorm()`: Probability function.
- `qinvnorm()`: Quantile function.
- `rinvnorm()`: Random generation.

Author(s)

David Gerard

References

- Robert, C. (1991). Generalized inverse normal distributions. Statistics & Probability Letters, 11(1), 37-41. doi:10.1016/01677152(91)90174P

Examples

```
x <- seq(-4, 4, length.out = 300)
y <- dinvnorm(x = x, imean = 0.1, isd = 1)
graphics::plot(x, y, type = "l")

p <- pinvnorm(q = x, imean = 0.1, isd = 1)
graphics::plot(x, p, type = "l", ylim = c(0, 1))
graphics::abline(h = c(0, 1), lty = 2)

qinvnorm(p = c(0.025, 0.5, 0.975), imean = 0.1, isd = 1)

s <- rinvnorm(n = 10000, imean = 0.1, isd = 1)
stats::quantile(s, c(0.025, 0.5, 0.975))
```

`mean_aug` *Augmented mean*

Description

If wt is an integer, this is mean(c(x, rep(A, wt))). But we generalize it to non-integer wt as well.

Usage

```
mean_aug(x, A, wt = 1)
```

Arguments

`x`	Data
`A`	Prior value
`wt`	Weight for A. Don't touch this unless you know what you are doing.

Value

The augmented mean

Author(s)

David Gerard

Examples

```
mean_aug(c(1, 2, 3), 4, wt = 10)
```

n1post *Posterior Distribution for n=1 Data*

Description

Density, probability, quantile, and random generation from the posterior distribution of the location parameter when one has $n = 1$ observation and uses a particular prior that results in valid confidence intervals (asymptotically in the confidence level).

Usage

```
nun1post(fam = c("normal", "cauchy"), ddist = NULL, ...)

dn1post(
  x,
  A,
  obs,
  nu = NULL,
  fam = c("normal", "cauchy"),
  ddist = NULL,
  log = FALSE,
  ...
)

pn1post(
  q,
  A,
  obs,
  nu = NULL,
  fam = c("normal", "cauchy"),
  pdist = NULL,
  lower.tail = TRUE,
  log.p = FALSE,
  ...
)

qn1post(
  p,
  A,
  obs,
  nu = NULL,
  fam = c("normal", "cauchy"),
  qdist = NULL,
  pdist = NULL,
  ...
)
```

```
rn1post(n, A, obs, nu = NULL, fam = c("normal", "cauchy"), rdist = NULL, ...)
```

Arguments

`fam`	One of `"normal"` or `"cauchy"`. If `ddist`, `pdist`, `qdist`, or `rdist` are specified then this argument is ignored.
`ddist`	Density function of standard distribution $\rho()$. Should have a `log` argument.
`...`	Additional arguments for `ddist`, `pdist`, `qdist`, and `rdist`.
`x, q`	vector of quantiles
`A`	The prior value.
`obs`	The observed value. This is X in the description.
`nu`	The prior value of nu. If not known, use `nun1post()`
`log, log.p`	logical; if `TRUE`, probabilities p are given as log(p).
`pdist`	Cumulative distribution function of standard distribution.
`lower.tail`	logical; if `TRUE` (default), probabilities are P(X<=x) otherwise, P(X>x).
`p`	vector of probabilities
`qdist`	Quantile function of standard distribution.
`n`	sample size
`rdist`	Random generation of standard distribution.

Details

Let X have PDF $\frac{1}{\sigma}\rho((x-\mu)/\sigma)$ for a known and symmetric $\rho(\cdot)$. Let $\nu = |\mu - A|/\sigma$ for a pre-specified A. We place prior $\pi(\mu) = |\mu - A|^{-1}$ and $\nu = \text{argmax}_{a>0} a\rho(a)$ with probability 1.

These functions will calculate the posterior distribution and allow you to interact with it through the distribution, density, quantile, and random generation functions.

Value

density, distribution, quantile, or random values.

Functions

- `nun1post()`: Obtains prior value of ν.
- `dn1post()`: Density function.
- `pn1post()`: Distribution function.
- `qn1post()`: Quantile function.
- `rn1post()`: Random generation.

Author(s)

David Gerard

Examples

```
set.seed(1)
# Observe x = 2, assume t with 2 df, prior value is A = 1
nun1post(ddist = dt, df = 2) ## nu should be 1
x <- seq(-10, 10, length.out = 500)
y <- dn1post(x = x, A = 1, obs = 2, nu = 1, ddist = dt, df = 2)
z <- rn1post(n = 10000, A = 1, obs = 2, nu = 1, rdist = rt, df = 2)
z <- z[z >= -10 & z <= 10]
graphics::hist(z, freq = FALSE, breaks = 200, border = "grey", col = "grey")
graphics::lines(x, y, type = "l", col = "#E69F00")
graphics::abline(v = c(1, 2), col = c("#56B4E9", "#009E73"), lty = c(2, 3))
pn1post(q = 2, A = 1, obs = 2, nu = 1, pdist = pt, df = 2)
qn1post(p = 0.7113, A = 1, obs = 2, nu = 1, qdist = qt, pdist = pt, df = 2)
```

---

p_wc — *Confidence distribution CDF based on n=1 confidence interval*

Description

Let $X \sim N(\mu, \sigma^2)$. Consider intervals of the form $X \pm \eta|X|$ or $X/2 \pm \eta|X|$. These intervals define a confidence distribution of μ. This is the CDF of the confidence distribution of $(\mu - X)/|X|$ or $(\mu - X/2)/|X|$. Please note that μ is random here not X. Also note that this confidence distribution does not exist between the 25th and 75th percentiles.

Usage

```
p_wc(q, center = c("X", "ave"))
```

Arguments

`q`	The quantile. Only defined for `abs(q) >= 1` when `center = "X"`, and for `abs(q) >= 0.5` when `center = "ave"`.
`center`	What is the center of the interval? Either `"X"` or `X/2`, (`"ave"`).

Value

n=1 confidence distribution CDF.

Author(s)

David Gerard

See Also

`d_wc()`

Examples

```
qseq <- seq(-10, 10, length.out = 100)
pseq <- p_wc(qseq)
plot(qseq, pseq, type = "l", xlab = "q", ylab = "CDF")
```

---

var_aug *Augmented variance*

Description

If wt is an integer, this is var(c(x, rep(A, wt))). But we generalize it to non-integer wt as well.

Usage

```
var_aug(x, A, wt = 1)
```

Arguments

x	Data
A	Prior value
wt	Weight for A. Don't touch this unless you know what you are doing.

Value

The augmented variance

Author(s)

David Gerard

Examples

```
var_aug(c(1, 2, 3), 4, wt = 10)
```

`wc_alpha` *Worst case exclusion probability of a n=1 confidence interval*

Description

This assumes the random variable X follows a normal distribution.

Usage

```
wc_alpha(width, center = c("X", "ave"))
```

Arguments

`width`	The half-width of the interval, in units of |X|. This needs to be at least 1 if `center = "X"`, or at least 0.5 if `center = "ave"`.
`center`	What is the center of the interval? Either `"X"` or `"ave"`.

Value

The worst case probability (a priori) that a confidence interval of half-width `width` will not capture the mean.

Author(s)

David Gerard

Examples

```
wc_alpha(width = 2, center = "X")
wc_alpha(width = 2, center = "ave")
```

`wc_cov` *Worst-case coefficient of variation*

Description

This assumes the random variable X is normal. Intervals are of the form center +/- width * |X|.

Usage

```
wc_cov(width, center = c("X", "ave"))
```

Arguments

`width`	The half-width of the interval, in units of |X|. This needs to be at least 1 if `center = "X"`, or at least 0.5 if `center = "ave"`.
`center`	What is the center of the interval? Either `"X"` or `"ave"`.

Details

We use A = 0 for calculations, but the worst case COV doesn't change if A is non-zero. So you can think of intervals of the form center +/- width * |X - A| where center is either X or (X + A)/2.

Value

Worst case coefficient of variation $\nu = |X - A|/\sigma$.

Author(s)

David Gerard

Examples

```
wc_cov(width = 2, center = "X")
wc_cov(width = 2, center = "ave")
```

`wc_width` *Given confidence level, provide half-width of CI*

Description

$(1-\alpha)100\%$ confidence intervals are of the form $X \pm \eta|X - A|$ or $(X + A)/2 \pm \eta|X - A|$. You give α and this function will provide η if X follows a Cauchy, normal, or uniform distribution.

Usage

```
wc_width(
  alpha,
  center = c("X", "ave", "approx"),
  family = c("normal", "cauchy", "uniform")
)
```

Arguments

`alpha`	The confidence error probability. We produce a (1-alpha)100 percent confidence interval.
`center`	Either X, ave, or the asymptotic width (approx).
`family`	Either the normal, Cauchy, or uniform distribution.

Value

Returns half-width of worst-case confidence interval in units of |X - A|.

Author(s)

David Gerard

Examples

```
wc_width(alpha = 0.2, center = "X")
wc_width(alpha = 0.2, center = "ave")
wc_width(alpha = 0.2, center = "approx")

wc_width(alpha = 0.05, center = "X")
wc_width(alpha = 0.05, center = "ave")
wc_width(alpha = 0.05, center = "approx")
```

Index