Package: smsncut (via r-universe)

May 19, 2026

Type Package

Title Optimal Diagnostic Cutoff Selection under Scale Mixtures of Skew-Normal Distributions

Version 0.1.0

Description Implements a parametric decision-theoretic framework for optimal diagnostic cutoff selection under the family of scale mixtures of skew-normal (SMSN) distributions, including the skew-normal (SN) and skew-t (ST) models as special cases. The optimal cutoff is defined by minimising a weighted misclassification risk that incorporates disease prevalence and asymmetric costs, leading to a likelihood-ratio equation that generalises the Youden criterion. Under a monotone likelihood ratio condition, existence, uniqueness, and global optimality of the cutoff are established. Asymptotic normality and a closed-form plug-in variance estimator are provided via the implicit function theorem and the multivariate delta method. Tools for model fitting, cutoff estimation, confidence intervals, the local identifiability diagnostic, and Monte Carlo simulation are included. The methodology is described in de Paula, Mouriño, and Dias Domingues (2026) <doi:10.48550/arXiv.2605.07829>.

License GPL-3

Encoding UTF-8

Depends R (>= 4.1.0)

Imports sn (>= 2.0.0), numDeriv (>= 2016.8-1)

Suggests testthat (>= 3.0.0), knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Renato de Paula [aut, cre] (ORCID: <`https://orcid.org/0000-0002-5835-5918`>), Helena Mouriño [aut], Tiago Dias Domingues [aut]

Maintainer Renato de Paula <`rrpaula@ciencias.ulisboa.pt`>

Config/roxygen2/version 8.0.0

Repository https://cran.r-universe.dev

Date/Publication 2026-05-19 08:50:02 UTC

RemoteUrl https://github.com/cran/smsncut

RemoteRef HEAD

RemoteSha df95c3f1321c3d9a33b2a0582fae0f6b1bc5c401

Contents

`cutoff_admissible_interval`

Construct the admissible interval [a, b]

Description

Builds a compact admissible interval for cutoff optimisation from the $(\alpha/2, 1-\alpha/2)$ quantiles of the fitted SMSN distributions (method `"effective_support"`) or from extreme quantiles of the pooled observed sample (method `"pooled_quantile"`).

Usage

```
cutoff_admissible_interval(
  theta0,
  theta1,
  family0,
  family1,
  method = c("effective_support", "pooled_quantile"),
  alpha = 0.01,
  x0 = NULL,
  x1 = NULL,
  probs = c(0.005, 0.995)
)
```

Arguments

`theta0`	Numeric vector. Parameters for group 0.
`theta1`	Numeric vector. Parameters for group 1.
`family0`	Character. Family for group 0.
`family1`	Character. Family for group 1.
`method`	Character. `"effective_support"` (default) or `"pooled_quantile"`.
`alpha`	Numeric in $(0, 1)$. Tail probability. Default `0.01`.
`x0`	Numeric vector. Group-0 data (required for `"pooled_quantile"`).
`x1`	Numeric vector. Group-1 data (required for `"pooled_quantile"`).
`probs`	Numeric vector of length 2. Quantile probabilities for `"pooled_quantile"`. Default `c(0.005, 0.995)`.

Value

Named numeric vector `c(a = ..., b = ...)`.

Examples

```
theta0 <- c(0, log(1), 1)
theta1 <- c(2, log(1), 1.5)
cutoff_admissible_interval(theta0, theta1, "SN", "SN")
```

`cutoff_boundary_check` *Verify boundary conditions of Proposition 5.7*

Description

Checks whether $\varphi(a;\theta) < 0$ and $\varphi(b;\theta) > 0$, which guarantees that the estimating equation $\varphi(c, \theta) = 0$ has a root inside (a, b).

Usage

```
cutoff_boundary_check(
  a,
  b,
  theta0,
  theta1,
  family0,
  family1,
  lambda0,
  lambda1,
  pi0,
  pi1
)
```

Arguments

`a`	Numeric. Left endpoint of the admissible interval.
`b`	Numeric. Right endpoint.
`theta0`	Numeric vector. Unconstrained parameters for group 0.
`theta1`	Numeric vector. Unconstrained parameters for group 1.
`family0`	Character. Family for group 0 (`"SN"` or `"ST"`).
`family1`	Character. Family for group 1 (`"SN"` or `"ST"`).
`lambda0`	Positive numeric. Cost of a false positive.
`lambda1`	Positive numeric. Cost of a false negative.
`pi0`	Numeric in $(0, 1)$. Prevalence of non-diseased.
`pi1`	Numeric in $(0, 1)$. Prevalence of diseased.

Value

A list with components `phi_a`, `phi_b`, `target_ratio`, `cond_a`, `cond_b`, `satisfied`.

Examples

```
theta0 <- c(0, log(1), 1)
theta1 <- c(2, log(1), 1.5)
ab   <- cutoff_admissible_interval(theta0, theta1, "SN", "SN")
cutoff_boundary_check(ab["a"], ab["b"], theta0, theta1, "SN", "SN",
                      1, 1, 0.5, 0.5)
```

`cutoff_build_interval` *Build a padded search interval for cutoff root-finding*

Description

Constructs an interval $[a-\delta\cdot w,\ b+\delta\cdot w]$ where $[a, b]$ is the 0.5th–99.5th percentile effective support of the two fitted distributions and $w = b - a$ is its width. This mirrors the interval construction in the Monte Carlo study of de Paula et al. (2026).

Usage

```
cutoff_build_interval(theta, family0, family1, pad = 0.5)
```

Arguments

`theta`	Numeric vector: `c(theta0, theta1)`.
`family0`	Character. Family for group 0.
`family1`	Character. Family for group 1.
`pad`	Non-negative numeric. Relative expansion factor δ. Default `0`.

Value

Named numeric vector `c(a = ..., b = ...)`, or `c(a = NA, b = NA)` if quantiles are non-finite.

Examples

```
theta <- c(0, log(1), 1, 2, log(1), 1.5)  # theta0 then theta1, both SN
cutoff_build_interval(theta, "SN", "SN")
cutoff_build_interval(theta, "SN", "SN", pad = 0.5)
```

`cutoff_ci` *Wald confidence interval for the optimal cutoff*

Description

Constructs a Wald-type $(1-\alpha)$ confidence interval for c^* (equation (31) of de Paula et al., 2026):

$$\hat{c} \pm z_{1-\alpha/2}\sqrt{\hat{V}/n}.$$

Usage

```
cutoff_ci(c_star, V, n, conf_level = 0.95)
```

Arguments

`c_star`	Numeric. The plug-in optimal cutoff.
`V`	Numeric. Asymptotic variance from `cutoff_variance`.
`n`	Integer. Total sample size $n = n_0 + n_1$.
`conf_level`	Numeric in $(0, 1)$. Default `0.95`.

Value

Named numeric vector `c(lower, upper, length)`.

Examples

```
cutoff_ci(c_star = 1.6, V = 0.81, n = 400, conf_level = 0.95)
```

`cutoff_expand_interval_until_valid`

Expand the search interval until boundary conditions hold

Description

Implements the additive interval-expansion strategy of de Paula et al. (2026, Section 6.1.1): starting from `pad` = `pad_init`, the padding is incremented by `pad_step` at each iteration until the boundary conditions $\varphi(a) < 0$ and $\varphi(b) > 0$ are satisfied, or `max_iter` attempts are exhausted.

Usage

```
cutoff_expand_interval_until_valid(
  theta,
  family0,
  family1,
  lambda0,
  lambda1,
  pi0,
  pi1,
  pad_init = 0,
  pad_step = 0.2,
  max_iter = 20L
)
```

Arguments

`theta`	Numeric vector: `c(theta0, theta1)`.
`family0`	Character. Family for group 0.
`family1`	Character. Family for group 1.
`lambda0`	Positive numeric. Cost of a false positive.

`lambda1`	Positive numeric. Cost of a false negative.
`pi0`	Numeric in $(0, 1)$. Prevalence of non-diseased.
`pi1`	Numeric in $(0, 1)$. Prevalence of diseased.
`pad_init`	Initial padding. Default `0`.
`pad_step`	Additive padding increment. Default `0.20`.
`max_iter`	Maximum expansion attempts. Default `20`.

Value

The output of `cutoff_boundary_check` for the final interval, augmented with a, b, ok, pad, and `iter`.

Examples

```
theta <- c(0, log(1), 1, 2, log(1), 1.5)
cutoff_expand_interval_until_valid(theta, "SN", "SN",
                                   lambda0 = 1, lambda1 = 1,
                                   pi0 = 0.5, pi1 = 0.5)
```

`cutoff_identifiability` *Local identifiability diagnostic*

Description

Computes $|\partial_c \varphi(c^*, \theta^*)|$ at the optimal cutoff (Remark 5.14 of de Paula et al., 2026). Small values indicate a locally flat estimating equation, inflated asymptotic variance, and suggest using bootstrap confidence intervals instead of Wald intervals.

Usage

```
cutoff_identifiability(
  c_star,
  theta0,
  theta1,
  family0,
  family1,
  lambda0,
  lambda1,
  pi0,
  pi1
)
```

Arguments

`c_star`	Numeric. The optimal cutoff.
`theta0`	Numeric vector. Unconstrained parameters for group 0.
`theta1`	Numeric vector. Unconstrained parameters for group 1.
`family0`	Character. Family for group 0 (`"SN"` or `"ST"`).
`family1`	Character. Family for group 1 (`"SN"` or `"ST"`).
`lambda0`	Positive numeric. Cost of a false positive.
`lambda1`	Positive numeric. Cost of a false negative.
`pi0`	Numeric in $(0, 1)$. Prevalence of non-diseased.
`pi1`	Numeric in $(0, 1)$. Prevalence of diseased.

Value

Positive numeric scalar.

Examples

```
theta0   <- c(0, log(1), 1)
theta1   <- c(2, log(1), 1.5)
c_star <- cutoff_optimal(theta0, theta1, "SN", "SN", 1, 1, 0.5, 0.5)$cutoff
cutoff_identifiability(c_star, theta0, theta1, "SN", "SN", 1, 1, 0.5, 0.5)
```

`cutoff_optimal` *Optimal diagnostic cutoff under SMSN distributions*

Description

Estimates the decision-theoretic optimal cutoff c^* as the minimiser of the weighted misclassification risk $R(c;\theta)$ over the admissible interval $[a, b]$ (Theorem 5.13 of de Paula et al., 2026). Under the monotone likelihood ratio condition, c^* is the unique root of $\varphi(c, \theta) = 0$, found by grid search followed by `uniroot`. If no root is found, `optimize` on R is used as fallback.

Usage

```
cutoff_optimal(
  theta0,
  theta1,
  family0,
  family1,
  lambda0,
  lambda1,
  pi0,
  pi1,
  interval = NULL,
  grid_length = 5000L
)
```

Arguments

`theta0`	Numeric vector. MLE for group 0.
`theta1`	Numeric vector. MLE for group 1.
`family0`	Character. Family for group 0.
`family1`	Character. Family for group 1.
`lambda0`	Positive numeric. Cost of a false positive.
`lambda1`	Positive numeric. Cost of a false negative.
`pi0`	Numeric in $(0, 1)$. Prevalence of non-diseased.
`pi1`	Numeric in $(0, 1)$. Prevalence of diseased.
`interval`	Numeric vector `c(a, b)`. If `NULL`, computed via `cutoff_admissible_interval`.
`grid_length`	Integer. Grid resolution for sign-change search. Default `5000`.

Value

A list with `cutoff`, `risk`, `interval`, and `boundary`.

Examples

```
theta0 <- c(0, log(1), 1)
theta1 <- c(2, log(1), 1.5)
res  <- cutoff_optimal(theta0, theta1, "SN", "SN", 1, 1, 0.5, 0.5)
res$cutoff
```

`cutoff_phi` *Estimating equation phi(c, theta)*

Description

Computes the derivative of the weighted misclassification risk with respect to the threshold (equation (15) of de Paula et al., 2026):

$$\varphi(c, \theta) = \lambda_1 \pi_1 f_1(c; \theta_1) - \lambda_0 \pi_0 f_0(c; \theta_0).$$

An interior minimiser satisfies $\varphi(c^*, \theta) = 0$, equivalently $\Lambda(c^*; \theta) = \lambda_0 \pi_0 / (\lambda_1 \pi_1)$.

Usage

```
cutoff_phi(c, theta0, theta1, family0, family1, lambda0, lambda1, pi0, pi1)
```

Arguments

`c`	Numeric scalar. Threshold value.
`theta0`	Numeric vector. Unconstrained parameters for group 0.
`theta1`	Numeric vector. Unconstrained parameters for group 1.
`family0`	Character. Family for group 0 ("`SN`" or "`ST`").
`family1`	Character. Family for group 1 ("`SN`" or "`ST`").
`lambda0`	Positive numeric. Cost of a false positive.
`lambda1`	Positive numeric. Cost of a false negative.
`pi0`	Numeric in $(0, 1)$. Prevalence of non-diseased.
`pi1`	Numeric in $(0, 1)$. Prevalence of diseased.

Value

Numeric scalar.

Examples

```
theta0 <- c(0, log(1), 1)
theta1 <- c(2, log(1), 1.5)
cutoff_phi(1.6, theta0, theta1, "SN", "SN", 1, 1, 0.5, 0.5)
```

`cutoff_risk` *Weighted misclassification risk*

Description

Computes the expected weighted misclassification risk at threshold c (equation (13) of de Paula et al., 2026):

$$R(c; \theta) = \lambda_1 \pi_1 F_1(c; \theta_1) + \lambda_0 \pi_0 [1 - F_0(c; \theta_0)].$$

Usage

```
cutoff_risk(c, theta0, theta1, family0, family1, lambda0, lambda1, pi0, pi1)
```

Arguments

`c`	Numeric scalar. Threshold value.
`theta0`	Numeric vector. Unconstrained parameters for group 0.
`theta1`	Numeric vector. Unconstrained parameters for group 1.
`family0`	Character. Family for group 0 ("`SN`" or "`ST`").
`family1`	Character. Family for group 1 ("`SN`" or "`ST`").
`lambda0`	Positive numeric. Cost of a false positive.
`lambda1`	Positive numeric. Cost of a false negative.
`pi0`	Numeric in $(0, 1)$. Prevalence of non-diseased.
`pi1`	Numeric in $(0, 1)$. Prevalence of diseased.

Value

Numeric scalar.

Examples

```
theta0 <- c(0, log(1), 1)
theta1 <- c(2, log(1), 1.5)
cutoff_risk(1.5, theta0, theta1, "SN", "SN", 1, 1, 0.5, 0.5)
```

`cutoff_variance` *Asymptotic variance of the plug-in cutoff estimator*

Description

Computes the closed-form plug-in asymptotic variance of $\hat{c}$ under separate SMSN maximum-likelihood fits (Theorem 5.13 and Proposition 5.15 of de Paula et al., 2026):

$$V = \frac{[\nabla_\theta \varphi]^\top \Sigma_{\mathrm{sep}} [\nabla_\theta \varphi]}{[\partial_c \varphi]^2},$$

where $\Sigma_{\mathrm{sep}} = n \cdot \mathrm{blockdiag}(H_0^{-1}, H_1^{-1})$ and H_k is the Hessian of the negative log-likelihood for group k (equation (26)).

Usage

```
cutoff_variance(
  c_star,
  theta0,
  theta1,
  family0,
  family1,
  lambda0,
  lambda1,
  pi0,
  pi1,
  hess0,
  hess1,
  n0,
  n1,
  ridge = 1e-06
)
```

Arguments

`c_star`	Numeric. The plug-in optimal cutoff.
`theta0`	Numeric vector. MLE for group 0.
`theta1`	Numeric vector. MLE for group 1.

`family0`	Character. Family for group 0.
`family1`	Character. Family for group 1.
`lambda0`	Positive numeric. Cost of a false positive.
`lambda1`	Positive numeric. Cost of a false negative.
`pi0`	Numeric in $(0, 1)$. Prevalence of non-diseased.
`pi1`	Numeric in $(0, 1)$. Prevalence of diseased.
`hess0`	Numeric matrix. Hessian of the negative log-likelihood for group 0 at the MLE (from `smsn_fit`).
`hess1`	Numeric matrix. Hessian for group 1.
`n0`	Integer. Group-0 sample size.
`n1`	Integer. Group-1 sample size.
`ridge`	Numeric. Ridge added to each Hessian diagonal for numerical stability. Default `1e-6`.

Value

Numeric scalar. The asymptotic variance V.

`cutoff_youden` *Parametric Youden cutoff under SMSN distributions*

Description

Estimates the Youden cutoff c_Y as the maximiser of $J(c;\theta) = F_0(c;\theta_0) - F_1(c;\theta_1)$ over the admissible interval (equation (9) of de Paula et al., 2026), implemented via `optimize` on $J(c)$ directly.

Usage

```
cutoff_youden(
  theta0,
  theta1,
  family0,
  family1,
  interval = NULL,
  grid_length = 5000L
)
```

Arguments

`theta0`	Numeric vector. Parameters for group 0.
`theta1`	Numeric vector. Parameters for group 1.
`family0`	Character. Family for group 0.
`family1`	Character. Family for group 1.
`interval`	Numeric vector of length 2. If `NULL`, computed automatically.
`grid_length`	Number of grid points used in the search.

Value

Numeric scalar. The estimated Youden cutoff.

Examples

```
theta0 <- c(0, log(1), 1)
theta1 <- c(2, log(1), 1.5)
cutoff_youden(theta0, theta1, "SN", "SN")
```

`mc_one_rep` *Run one Monte Carlo replication*

Description

Run one Monte Carlo replication

Usage

```
mc_one_rep(
  theta0_true,
  theta1_true,
  family0,
  family1,
  c_true,
  lambda0,
  lambda1,
  pi0,
  pi1,
  n0,
  n1,
  flat_tol = 0.001,
  conf_level = 0.95
)
```

Arguments

`theta0_true`	True parameter vector for group 0.
`theta1_true`	True parameter vector for group 1.
`family0`	Family for group 0.
`family1`	Family for group 1.
`c_true`	True optimal cutoff.
`lambda0`	Cost of a false positive.
`lambda1`	Cost of a false negative.
`pi0`	Prevalence of group 0.
`pi1`	Prevalence of group 1.

`n0`	Sample size for group 0.
`n1`	Sample size for group 1.
`flat_tol`	Minimum admissible value of the identifiability diagnostic.
`conf_level`	Confidence level.

Value

A list describing the replication outcome.

`mc_run_all` *Run all scenarios*

Description

Run all scenarios

Usage

```
mc_run_all(
  scenarios,
  n_vec = c(200L, 400L, 1000L),
  B = 2000L,
  flat_tol = 0.001,
  conf_level = 0.95,
  seed = NULL
)
```

Arguments

`scenarios`	Named list of scenarios.
`n_vec`	Vector of total sample sizes.
`B`	Number of replications.
`flat_tol`	Identifiability tolerance.
`conf_level`	Confidence level.
`seed`	Optional random seed.

Value

Named list of simulation outputs.

`mc_simulate_scenario` *Monte Carlo simulation for one scenario*

Description

Monte Carlo simulation for one scenario

Usage

```
mc_simulate_scenario(
  theta0_true,
  theta1_true,
  family0,
  family1,
  c_true,
  lambda0,
  lambda1,
  pi0,
  pi1,
  n0,
  n1,
  B = 2000L,
  flat_tol = 0.001,
  conf_level = 0.95,
  seed = NULL
)
```

Arguments

`theta0_true`	True parameter vector for group 0.
`theta1_true`	True parameter vector for group 1.
`family0`	Family for group 0.
`family1`	Family for group 1.
`c_true`	True optimal cutoff.
`lambda0`	Cost of a false positive.
`lambda1`	Cost of a false negative.
`pi0`	Prevalence of group 0.
`pi1`	Prevalence of group 1.
`n0`	Sample size for group 0.
`n1`	Sample size for group 1.
`B`	Number of replications.
`flat_tol`	Minimum admissible value of the identifiability diagnostic.
`conf_level`	Confidence level.
`seed`	Optional random seed.

Value

A list of Monte Carlo summary statistics.

`smsn_auc` *Plug-in AUC estimator under SMSN distributions*

Description

Approximates the AUC by the trapezoidal rule with `h` grid points (equation (13) of de Paula et al., 2026). The approximation error is $O(h^{-2})$; the default $h = 1000$ gives error $\approx 10^{-6}$.

Usage

```
smsn_auc(theta0, theta1, family0, family1, h = 1000L)
```

Arguments

`theta0`	Numeric vector. Parameters for group 0.
`theta1`	Numeric vector. Parameters for group 1.
`family0`	Character. Family for group 0.
`family1`	Character. Family for group 1.
`h`	Integer. Number of grid points. Default `1000`.

Value

Numeric scalar in $[0, 1]$.

Examples

```
theta0 <- c(0, log(1), 1)
theta1 <- c(2, log(1), 1.5)
smsn_auc(theta0, theta1, "SN", "SN")
```

`smsn_cdf` *CDF of an SMSN distribution*

Description

Evaluates the cumulative distribution function of a skew-normal (SN) or skew-t (ST) distribution at the unconstrained parametrisation $\theta = (\xi, \log\omega, \alpha)$ for SN, or $\theta = (\xi, \log\omega, \alpha, \log(\nu - 2))$ for ST.

Usage

```
smsn_cdf(x, theta, family = c("SN", "ST"))
```

Arguments

`x`	Numeric vector of evaluation points.
`theta`	Numeric vector of unconstrained parameters. Length 3 for SN, length 4 for ST.
`family`	Character string, either `"SN"` or `"ST"`.

Value

Numeric vector of CDF values in $[0, 1]$.

Examples

```
theta_sn <- c(0, log(1), 1.5)
smsn_cdf(c(-1, 0, 1, 2), theta_sn, "SN")
```

`smsn_density` *Density of an SMSN distribution*

Description

Evaluates the density of a skew-normal (SN) or skew-t (ST) distribution at the unconstrained parametrisation $\theta = (\xi, \log\omega, \alpha)$ for SN, or $\theta = (\xi, \log\omega, \alpha, \log(\nu - 2))$ for ST.

Usage

```
smsn_density(x, theta, family = c("SN", "ST"))
```

Arguments

`x`	Numeric vector of evaluation points.
`theta`	Numeric vector of unconstrained parameters. Length 3 for SN, length 4 for ST.
`family`	Character string, either `"SN"` or `"ST"`.

Value

Numeric vector of density values.

Examples

```
theta_sn <- c(0, log(1), 1.5)
smsn_density(c(-1, 0, 1, 2), theta_sn, "SN")

theta_st <- c(0, log(1), 1.5, log(8 - 2))
smsn_density(c(-1, 0, 1, 2), theta_st, "ST")
```

`smsn_fit` *Fit an SMSN model by maximum likelihood*

Description

Fits a skew-normal (SN) or skew-t (ST) distribution to a univariate sample by maximising the log-likelihood, using the unconstrained parametrisation $\theta = (\xi, \log\omega, \alpha)$ for SN or $\theta = (\xi, \log\omega, \alpha, \log(\nu-2))$ for ST.

Usage

```
smsn_fit(x, family = c("SN", "ST"), maxit = 4000L, reltol = 1e-10)
```

Arguments

`x`	Numeric vector of observations. Non-finite values are removed.
`family`	Character string, either `"SN"` (default) or `"ST"`.
`maxit`	Integer. Maximum number of iterations for `optim`. Default 4000.
`reltol`	Numeric. Relative convergence tolerance. Default `1e-10`.

Details

The observed Fisher information matrix is computed via Richardson extrapolation as implemented in `numDeriv::hessian()`, which is critical for the reliability of the plug-in variance estimator $\hat{V}$ in heavy-tailed models (de Paula et al., 2026, Section 6.2).

Value

A list with components:

`par`	MLE in unconstrained parametrisation θ.
`hessian`	Hessian of the negative log-likelihood at the MLE, computed by Richardson extrapolation.
`logLik`	Maximised log-likelihood.
`family`	Fitted family (`"SN"` or `"ST"`).
`convergence`	Convergence code from `optim` (0 = success).

References

de Paula, R., Mouriño, H., and Dias Domingues, T. (2026). Parametric ROC analysis and optimal cutoff selection under scale mixtures of skew-normal distributions: a decision-theoretic framework with asymptotic inference. *arXiv:2605.07829*. doi:10.48550/arXiv.2605.07829.

Examples

```
set.seed(1)
x <- sn::rsn(200, xi = 0, omega = 1, alpha = 2)
fit <- smsn_fit(x, family = "SN")
fit$par
fit$convergence
```

`smsn_quantile` *Quantile function of an SMSN distribution*

Description

Returns quantiles of a skew-normal (SN) or skew-t (ST) distribution at the unconstrained parametrisation $\theta = (\xi, \log\omega, \alpha)$ for SN, or $\theta = (\xi, \log\omega, \alpha, \log(\nu - 2))$ for ST.

Usage

```
smsn_quantile(prob, theta, family = c("SN", "ST"))
```

Arguments

`prob`	Numeric vector of probabilities in $(0, 1)$.
`theta`	Numeric vector of unconstrained parameters. Length 3 for SN, length 4 for ST.
`family`	Character string, either `"SN"` or `"ST"`.

Value

Numeric vector of quantiles.

Examples

```
theta_sn <- c(0, log(1), 1.5)
smsn_quantile(c(0.025, 0.5, 0.975), theta_sn, "SN")
```

`smsn_rdist` *Random generation from an SMSN distribution*

Description

Generates random observations from a skew-normal (SN) or skew-t (ST) distribution under the unconstrained parametrisation $\theta = (\xi, \log \omega, \alpha)$ for SN, or $\theta = (\xi, \log \omega, \alpha, \log(\nu - 2))$ for ST.

Usage

```
smsn_rdist(n, theta, family = c("SN", "ST"))
```

Arguments

`n`	Integer. Number of observations.
`theta`	Numeric vector of unconstrained parameters. Length 3 for SN, length 4 for ST.
`family`	Character string, either `"SN"` or `"ST"`.

Value

Numeric vector of length `n`.

Examples

```
set.seed(1)
theta_sn <- c(0, log(1), 1.5)
x <- smsn_rdist(100L, theta_sn, "SN")
hist(x, breaks = 20, main = "SN sample")
```

`smsn_roc` *Parametric ROC curve under SMSN distributions*

Description

Evaluates the parametric ROC curve at false positive rates r (equation (6) of de Paula et al., 2026):

$$\mathrm{ROC}(r; \theta) = 1 - F_1(Q_0(1 - r; \theta_0); \theta_1).$$

Usage

```
smsn_roc(r, theta0, theta1, family0, family1)
```

Arguments

`r`	Numeric vector of false positive rates in $(0, 1)$.
`theta0`	Numeric vector. Parameters for group 0.
`theta1`	Numeric vector. Parameters for group 1.
`family0`	Character. Family for group 0.
`family1`	Character. Family for group 1.

Value

Numeric vector of true positive rates.

Examples

```
theta0 <- c(0, log(1), 1)
theta1 <- c(2, log(1), 1.5)
r    <- seq(0.01, 0.99, by = 0.01)
plot(r, smsn_roc(r, theta0, theta1, ”SN”, ”SN”), type = ”l”,
     xlab = ”FPR”, ylab = ”TPR”)
abline(0, 1, lty = 2)
```

Index