Package 'DropCens'

August 9, 2026

Type Package

Title Distributionally Robust Progressive Type-II Censoring Inference

Version 0.1.0

Description Implements distributionally robust optimization algorithms for progressive Type-II censored Weibull lifetime data under Wasserstein ambiguity balls. Provides primal-dual convex optimization solvers and robust inference bounds.

License GPL-3

Encoding UTF-8

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author Okechukwu J. Obulezi [aut, cre]

Maintainer Okechukwu J. Obulezi <oj.obulezi@unizik.edu.ng>

Repository CRAN

Date/Publication 2026-08-09 08:00:17 UTC

Contents

`dro_bayesian_bounds` *Wasserstein-Bounded Robust Bayesian Bounds*

Description

Computes upper and lower robust inference bounds across extreme prior profiles.

Usage

```
dro_bayesian_bounds(t, R, delta = 0.1)
```

Arguments

t	Numeric vector of observed failure/censoring times.
R	Numeric vector of progressive removal counts.
delta	Numeric prior ambiguity radius (default: 0.1).

Value

Matrix of lower and upper robust credible bounds.

Examples

```
t_obs <- sort(rweibull(10, shape = 2, scale = 8))
R_obs <- rep(0, 10)
bounds <- dro_bayesian_bounds(t = t_obs, R = R_obs, delta = 0.05)
print(bounds)
```

---

fit_dro_pc *Primal-Dual Convex Optimization Solver for DRO-PC Estimator*

Description

Solves worst-case distributionally robust MLE parameter estimates under Wasserstein ambiguity balls for progressive Type-II censored Weibull data.

Usage

```
fit_dro_pc(t, R, epsilon = 0.05, max_iter = 500, tol = 1e-05)
```

Arguments

t	Numeric vector of observed failure/censoring times.
R	Numeric vector of progressive removal counts.
epsilon	Numeric radius of the Wasserstein ambiguity ball (default: 0.05).
max_iter	Integer maximum number of iterations (default: 500).
tol	Numeric convergence tolerance (default: 1e-5).

Value

A list containing estimated robust parameters, dual multiplier, and covariance.

Examples

```
set.seed(123)
t_obs <- sort(rweibull(15, shape = 2, scale = 10))
R_obs <- c(2, rep(0, 13), 3)
fit <- fit_dro_pc(t = t_obs, R = R_obs, epsilon = 0.02)
print(fit$par)
```

Index