

Package ‘rinet’

October 20, 2026

Type Package

Title Clinical Reference Interval Estimation with Reference Interval Network (RINet)

Version 0.2.0

Description Predicts statistics of a reference distribution from a mixture of raw clinical measurements (healthy and pathological). Uses pretrained CNN models to estimate the mean, standard deviation, and reference fraction from 1D or 2D sample data. Methods are described in LeBien, Velez, and Roche-Lima (2026) ``RINet: synthetic data training for indirect estimation of clinical reference distributions" <doi:10.1016/j.jbi.2026.104980>.

License MIT + file LICENSE

URL https://github.com/jackgle/rinet_r_package

BugReports https://github.com/jackgle/rinet_r_package/issues

Encoding UTF-8

Depends R (>= 3.5.0)

Imports torch, stats

Config/roxygen2/version 8.1.1

NeedsCompilation no

Author Jack LeBien [aut, cre]

Maintainer Jack LeBien <jlebien@abartyshealth.com>

Repository CRAN

Date/Publication 2026-10-20 13:20:08 UTC

Contents

predict_rinet	2
predict_rinet_1d	4
predict_rinet_2d	5
Index	8

predict_rinet	<i>Predict statistics of the underlying reference distribution from mixture distributions using RINet</i>
---------------	---

Description

Automatically detects whether input data is 1D or 2D and calls the appropriate prediction function. This is the main user-facing function. It estimates the statistics of a "healthy" reference population from a mixture of healthy and pathological measurements.

Usage

```
predict_rinet(
  data,
  feature_grid_range = c(-4, 4),
  feature_grid_nbins = 100,
  verbose = 0,
  log_scale = TRUE,
  percentiles = c(0.025, 0.975),
  n_bootstrap = 0,
  confidence_level = 0.95,
  model_version = c("v2.1", "v2.0")
)
```

Arguments

data	A numeric vector, matrix, or list. For 1D: vector or matrix with 1 column. For 2D: matrix with 2 columns. Can also be a list of such objects.
feature_grid_range	Numeric vector of length 2 specifying the range for histogram binning. Default is c(-4, 4).
feature_grid_nbins	Integer specifying the number of histogram bins. Default is 100.
verbose	Integer controlling verbosity (0 = silent). Default is 0.
log_scale	Logical indicating whether to log-transform the data before prediction. If TRUE (default), returns log-scale statistics and calculates reference intervals in the original scale. Default is TRUE.
percentiles	Numeric vector of length 2 specifying the lower and upper percentiles for the reference interval. Default is c(0.025, 0.975).
n_bootstrap	Integer specifying the number of bootstrap resamples for confidence intervals. Default is 0 (no bootstrap). When > 0, confidence intervals are computed for all predicted statistics using batch inference.
confidence_level	Numeric specifying the confidence level for bootstrap intervals. Default is 0.95.

`model_version` Version of the 1D model: "v2.1" (default) or "v2.0", the model described in the published paper. Only applies to 1D data; 2D predictions always use the v2.0 model. See the package NEWS for the differences between versions.

Value

A list of predictions. Each element contains:

<code>mean</code>	Predicted mean(s) (log-scale if <code>log_scale=TRUE</code>)
<code>std</code>	Predicted standard deviation(s) (log-scale if <code>log_scale=TRUE</code>)
<code>covariance</code>	Predicted covariance matrix
<code>correlation</code>	Predicted correlation (NA for 1D)
<code>reference_fraction</code>	Predicted reference component fraction
<code>reference_interval</code>	Reference interval in original scale (if <code>log_scale=TRUE</code>)
<code>log_scale</code>	Logical indicating whether log-scaling was used
<code>model_version</code>	Version of the model that produced the prediction
<code>bootstrap_ci</code>	List of bootstrap confidence intervals (if <code>n_bootstrap > 0</code>)

Setup

The models run on the torch backend, which is downloaded separately from the package. Run `torch::install_torch()` once after installing `rinet`.

Examples

```
## Not run:
# 1D sample (using positive data for log-scale)
sample_1d <- exp(rnorm(1000, mean = 2, sd = 0.5))
result <- predict_rinet(sample_1d)

# 2D sample (using positive data for log-scale)
sample_2d <- exp(matrix(rnorm(2000, mean = 2, sd = 0.5), ncol = 2))
result <- predict_rinet(sample_2d)

# Multiple samples (automatically detected)
samples <- list(exp(rnorm(1000, mean = 2, sd = 0.5)),
               exp(rnorm(1000, mean = 2, sd = 0.5)))
results <- predict_rinet(samples)

# Reproduce the published paper's 1D model
result <- predict_rinet(sample_1d, model_version = "v2.0")

## End(Not run)
```

predict_rinet_1d	<i>Predict statistics of the underlying reference distribution from 1D mixture distributions using RINet</i>
------------------	--

Description

Takes one or more 1D samples and predicts the underlying reference population statistics (mean, std, reference fraction) from a mixture of healthy and pathological measurements.

Usage

```
predict_rinet_1d(
  data,
  feature_grid_range = c(-4, 4),
  feature_grid_nbins = 100,
  verbose = 0,
  log_scale = TRUE,
  percentiles = c(0.025, 0.975),
  n_bootstrap = 0,
  confidence_level = 0.95,
  model_version = c("v2.1", "v2.0")
)
```

Arguments

data	A numeric vector, matrix, or list of vectors. Each sample should contain observations from a 1D mixture distribution.
feature_grid_range	Numeric vector of length 2 specifying the range for histogram binning. Default is c(-4, 4).
feature_grid_nbins	Integer specifying the number of histogram bins. Default is 100.
verbose	Integer controlling verbosity (0 = silent). Default is 0.
log_scale	Logical indicating whether to log-transform the data before prediction. If TRUE (default), returns log-scale statistics and calculates reference intervals in the original scale. Default is TRUE.
percentiles	Numeric vector of length 2 specifying the lower and upper percentiles for the reference interval. Default is c(0.025, 0.975).
n_bootstrap	Integer specifying the number of bootstrap resamples for confidence intervals. Default is 0 (no bootstrap). When > 0, confidence intervals are computed for all predicted statistics.
confidence_level	Numeric specifying the confidence level for bootstrap intervals. Default is 0.95.
model_version	Version of the 1D model: "v2.1" (default) or "v2.0", the model described in the published paper. See the package NEWS for the differences between versions.

Value

A list of predictions. Each element contains:

mean	Predicted mean (scalar, log-scale if log_scale=TRUE)
std	Predicted standard deviation (scalar, log-scale if log_scale=TRUE)
covariance	Covariance matrix (1x1 matrix)
correlation	Always NA for 1D
reference_fraction	Predicted reference component fraction
reference_interval	Reference interval in original scale (if log_scale=TRUE)
log_scale	Logical indicating whether log-scaling was used
model_version	Version of the model that produced the prediction
bootstrap_ci	List of bootstrap confidence intervals (if n_bootstrap > 0): mean_ci, std_ci, reference_fraction_ci, reference_interval_lower_ci, reference_interval_upper_ci

Examples

```
## Not run:
# Single sample (using positive data for log-scale)
sample1 <- exp(rnorm(1000, mean = 2, sd = 0.3))
result <- predict_rinet_1d(sample1)
print(result[[1]]$mean)

# Multiple samples
samples <- list(exp(rnorm(1000, 2, 0.3)), exp(rnorm(1000, 1.5, 0.4)))
results <- predict_rinet_1d(samples)

## End(Not run)
```

predict_rinet_2d	<i>Predict statistics of the underlying reference distribution from 2D mixture distributions using RINet</i>
------------------	--

Description

Takes one or more 2D samples and predicts the underlying reference population statistics (means, stds, correlation, reference fraction) from a mixture of healthy and pathological measurements.

Usage

```
predict_rinet_2d(
  data,
  feature_grid_range = c(-4, 4),
  feature_grid_nbins = 100,
  verbose = 0,
```

```

    log_scale = TRUE,
    percentiles = c(0.025, 0.975),
    n_bootstrap = 0,
    confidence_level = 0.95
  )

```

Arguments

<code>data</code>	A matrix or list of matrices. Each sample should be a matrix with 2 columns representing observations from a 2D mixture distribution.
<code>feature_grid_range</code>	Numeric vector of length 2 specifying the range for histogram binning. Default is <code>c(-4, 4)</code> .
<code>feature_grid_nbins</code>	Integer specifying the number of histogram bins. Default is 100.
<code>verbose</code>	Integer controlling verbosity (0 = silent). Default is 0.
<code>log_scale</code>	Logical indicating whether to log-transform the data before prediction. If TRUE (default), returns log-scale statistics and calculates reference intervals in the original scale. Default is TRUE.
<code>percentiles</code>	Numeric vector of length 2 specifying the lower and upper percentiles for the reference interval. Default is <code>c(0.025, 0.975)</code> .
<code>n_bootstrap</code>	Integer specifying the number of bootstrap resamples for confidence intervals. Default is 0 (no bootstrap). When > 0, confidence intervals are computed for all predicted statistics.
<code>confidence_level</code>	Numeric specifying the confidence level for bootstrap intervals. Default is 0.95.

Value

A list of predictions. Each element contains:

<code>mean</code>	Predicted means (vector of length 2, log-scale if <code>log_scale=TRUE</code>)
<code>std</code>	Predicted standard deviations (vector of length 2, log-scale if <code>log_scale=TRUE</code>)
<code>covariance</code>	Predicted covariance matrix (2x2 matrix)
<code>correlation</code>	Predicted correlation coefficient (scalar)
<code>reference_fraction</code>	Predicted reference component fraction
<code>reference_interval</code>	Reference region ellipse vertices (100x2 matrix) in original scale (if <code>log_scale=TRUE</code>)
<code>log_scale</code>	Logical indicating whether log-scaling was used
<code>model_version</code>	Version of the model that produced the prediction (always "v2.0" for 2D)
<code>bootstrap_ci</code>	List of bootstrap confidence intervals (if <code>n_bootstrap > 0</code>): <code>mean_ci</code> (2x2 matrix), <code>std_ci</code> (2x2 matrix), <code>correlation_ci</code> , <code>reference_fraction_ci</code>

Examples

```
## Not run:
# Single 2D sample (using positive data for log-scale)
sample1 <- exp(matrix(rnorm(2000, mean = 2, sd = 0.3), ncol = 2))
result <- predict_rinet_2d(sample1)
print(result[[1]]$mean)
print(result[[1]]$covariance)

# Multiple samples
samples <- list(exp(matrix(rnorm(2000, mean = 2, sd = 0.3), ncol = 2)),
                exp(matrix(rnorm(2000, mean = 2, sd = 0.3), ncol = 2)))
results <- predict_rinet_2d(samples)

## End(Not run)
```

Index

`predict_rinet`, [2](#)
`predict_rinet_1d`, [4](#)
`predict_rinet_2d`, [5](#)