Package 'depthR'

June 26, 2026

Type Package

Title Multivariate Depth Functions for General Dimension

Version 0.1.8

Date 2026-06-22

Description Efficient computation of multivariate statistical depth functions in arbitrary dimension d. Implements Mahalanobis depth, Tukey (halfspace) depth, Liu simplicial depth (via adaptive Monte Carlo), projection depth, and spatial depth. Provides depth-based medians, central regions, outlier detection, and depth-depth plots. 'C++' backends via 'Rcpp' and 'RcppEigen' ensure performance at large n and d.
References: Liu (1990) <doi:10.1214/aos/1176347507>, Zuo and Serfling (2000) <doi:10.1214/aos/1016218226>, Vardi and Zhang (2000) <doi:10.1073/pnas.97.4.1423>.

License GPL (>= 3)

URL https://github.com/penny4nonsense/depthR

BugReports https://github.com/penny4nonsense/depthR/issues

Depends R (>= 3.1.0)

Imports Rcpp (>= 1.0.0), RcppParallel (>= 5.0.0), stats, graphics

LinkingTo Rcpp, RcppEigen (>= 0.3.3), RcppParallel (>= 5.0.0)

Suggests testthat (>= 3.0.0), MASS, knitr, rmarkdown

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

Config/roxygen2/version 8.0.0

RoxygenNote 7.3.3

NeedsCompilation yes

Author Jason Parker [aut, cre]

Maintainer Jason Parker <jparker588@gmail.com>

Repository CRAN

Date/Publication 2026-06-26 09:40:02 UTC

Contents

`central_region` *Depth-Based Central Region*

Description

Returns the set of observations whose depth is at or above the `alpha`-th quantile of the depth distribution — the multivariate analog of a quantile interval.

Usage

```
central_region(x, alpha = 0.5, ...)
```

Arguments

`x`	A `depth` object from `compute_depth()`.
`alpha`	Numeric scalar in (0, 1). The central region contains the deepest `1 - alpha` fraction of observations. Default 0.50 (the inner half).
`...`	Ignored.

Value

A named list:

indices Row indices of observations in the central region.

points Matrix of observations in the central region.

depths Depth values of those observations.

threshold The depth cutoff used.

alpha The alpha level used.

compute_depth *Compute Depth*

Description

Computes the statistical depth of every row of `data` with respect to the empirical distribution of `data`, returning a `depth` object from which medians, outliers, ranks, and other derived quantities can be extracted cheaply without recomputing depth.

Usage

```
compute_depth(data, depth_fn = mahalanobis_depth, ...)
```

Arguments

`data`	Numeric matrix (n x d) or data frame. Rows are observations, columns are variables.
`depth_fn`	Depth function to use. Must have signature `f(x, data, ...)` and return a numeric vector of length `nrow(x)`. Defaults to `mahalanobis_depth`.
`...`	Additional arguments forwarded to `depth_fn`.

Value

An object of class `"depth"` with components:

depths Numeric vector of length n — depth of each observation.

data The original data matrix.

depth_fn The depth function used.

n Number of observations.

d Dimension.

call The matched call.

Examples

```
set.seed(42)
data <- matrix(rnorm(500), nrow = 100, ncol = 5)
dd   <- compute_depth(data, depth_fn = mahalanobis_depth)

median(dd)
rank(dd)
outliers(dd)
summary(dd)
plot(dd)
```

dd_plot *Depth-Depth Plot*

Description

Computes and plots the depth-depth (DD) plot for two samples. Each observation from both samples is assigned two depth values — its depth with respect to the empirical distribution of x and its depth with respect to the empirical distribution of y. Points from the same distribution cluster near the main diagonal.

Usage

```
dd_plot(
  x,
  y,
  depth_fn = simplicial_depth,
  plot = TRUE,
  xlab = "Depth wrt X",
  ylab = "Depth wrt Y",
  main = "DD-Plot",
  col_x = "steelblue",
  col_y = "firebrick",
  pch_x = 19L,
  pch_y = 17L,
  legend = TRUE,
  ...
)
```

Arguments

x	Numeric matrix (n1 x d) — first sample.
y	Numeric matrix (n2 x d) — second sample. Must have the same number of columns as x.
depth_fn	Depth function to use. Must have signature f(x, data, ...). Defaults to simplicial_depth.
plot	Logical. If TRUE (default), produce the plot.
xlab	Label for the x-axis. Defaults to "Depth wrt X".
ylab	Label for the y-axis. Defaults to "Depth wrt Y".
main	Plot title. Defaults to "DD-Plot".
col_x	Color for points from x. Default "steelblue".
col_y	Color for points from y. Default "firebrick".
pch_x	Plot character for points from x. Default 19.
pch_y	Plot character for points from y. Default 17.
legend	Logical. If TRUE (default), add a legend.
...	Additional arguments passed to depth_fn.

Details

The DD-plot was introduced by Liu, Parelius & Singh (1999) as a nonparametric graphical tool for two-sample comparison. It is the multivariate analog of the QQ-plot, using depth in place of quantiles.

If the two distributions are identical, all points should fall near the diagonal. Systematic deviations indicate location shifts (points above or below the diagonal) or scale/shape differences (spread of points away from the diagonal).

Value

Invisibly returns a data frame with columns:

depth_x Depth of each observation with respect to `x`.

depth_y Depth of each observation with respect to `y`.

sample Factor indicating which sample the observation came from.

References

Liu, R. Y., Parelius, J. M. & Singh, K. (1999). Multivariate analysis by data depth: descriptive statistics, graphics and inference. *Annals of Statistics*, 27(3), 783–858.

Examples

```
set.seed(42)
# Same distribution — points near diagonal
x <- matrix(rnorm(200), nrow = 100, ncol = 2)
y <- matrix(rnorm(200), nrow = 100, ncol = 2)
dd_plot(x, y, depth_fn = simplicial_depth)

# Location shift — points systematically off diagonal
y_shift <- matrix(rnorm(200, mean = 1), nrow = 100, ncol = 2)
dd_plot(x, y_shift, depth_fn = tukey_depth)

# Store results without plotting
result <- dd_plot(x, y, plot = FALSE)
head(result)
```

---

depth_outlyingness *Depth-Based Outlyingness*

Description

Converts depth values to outlyingness scores via O(x) = 1/D(x) - 1, so that depth 1 maps to outlyingness 0 and depth approaching 0 maps to outlyingness approaching infinity.

Usage

```
depth_outlyingness(depths)
```

Arguments

depths	Numeric vector of depth values in (0, 1].

Value

Numeric vector of outlyingness values in [0, inf).

---

mahalanobis_depth *Mahalanobis Depth*

Description

Computes the Mahalanobis depth of one or more query points with respect to a reference distribution estimated from data.

Usage

```
mahalanobis_depth(x, data, mu = NULL, sigma = NULL)
```

Arguments

x	Numeric matrix of query points (m x d), or a numeric vector of length d for a single query point.
data	Numeric matrix of reference data (n x d). Used to estimate the mean and covariance.
mu	Optional numeric vector of length d. If supplied, overrides the mean estimated from data.
sigma	Optional numeric matrix (d x d). If supplied, overrides the covariance estimated from data. Must be positive definite.

Value

Numeric vector of depth values in (0, 1], one per query point.

median *Median*

Description

Generic function for computing the median. For `depth` objects, returns the deepest observation. For all other objects, delegates to `stats::median`.

Usage

```
median(x, ...)
```

Arguments

`x`	An object. For `depth` objects, see `median.depth`.
`...`	Additional arguments passed to methods.

Value

For `depth` objects, a named list with elements `point`, `depth`, and `index`. For other objects, see `median`.

median.depth *Depth-Based Median*

Description

Returns the observation with the highest depth — the multivariate analog of the median.

Usage

```
## S3 method for class 'depth'
median(x, ...)
```

Arguments

`x`	A `depth` object from `compute_depth()`.
`...`	Ignored.

Value

A named list:

point Numeric vector of length d — the deepest observation.

depth Depth value at the median.

index Row index of the deepest observation in the data.

`outliers` *Depth-Based Outlier Detection*

Description

Flags observations whose depth falls below a threshold as outliers. The threshold can be specified as a quantile of the depth distribution (default) or as an absolute depth cutoff.

Usage

```
outliers(x, threshold = 0.05, absolute = FALSE, ...)
```

Arguments

`x`	A `depth` object from `compute_depth()`.
`threshold`	Numeric scalar in (0, 1). Interpreted as a quantile of the depth distribution when `absolute = FALSE` (default): the bottom `threshold` fraction of observations are flagged as outliers. When `absolute = TRUE`, any observation with depth below `threshold` is flagged.
`absolute`	Logical. If `TRUE`, `threshold` is an absolute depth cutoff rather than a quantile. Default `FALSE`.
`...`	Ignored.

Value

A named list:

outlier Logical vector of length n — `TRUE` for outliers.

indices Integer vector of row indices of outlying observations.

points Matrix of outlying observations.

depths Depth values of outlying observations.

threshold The actual depth cutoff used.

Examples

```
set.seed(42)
data <- matrix(rnorm(500), nrow = 100, ncol = 5)
dd   <- compute_depth(data)

# Flag bottom 5% by depth (default)
outliers(dd)

# Flag bottom 10%
outliers(dd, threshold = 0.10)

# Absolute depth cutoff
```

```
outliers(dd, threshold = 0.05, absolute = TRUE)
```

`plot.depth` *Plot a Depth Object*

Description

For d = 2, plots the data with point size proportional to depth and outliers flagged in red. For d > 2, plots a depth profile (observation index vs depth value).

Usage

```
## S3 method for class 'depth'
plot(x, outlier_threshold = 0.05, main = NULL, ...)
```

Arguments

`x`	A `depth` object from `compute_depth()`.
`outlier_threshold`	Quantile threshold for flagging outliers. Default 0.05.
`main`	Plot title. If `NULL` (default), a sensible title is generated automatically.
`...`	Additional arguments passed to `plot()`.

Value

Invisibly returns `x`, the original `depth` object. Called primarily for its side effect of producing a plot.

`projection_depth` *Projection Depth*

Description

Computes the projection depth of one or more query points with respect to a reference distribution estimated from `data`, using an adaptive random projection approximation with parallel computation.

Usage

```
projection_depth(
  x,
  data,
  tol = 0.01,
  batch_size = 100L,
  min_batches = 5L,
  patience = 3L,
  seed = 42L
)
```

Arguments

x	Numeric matrix of query points (m x d), or a numeric vector of length d for a single point.
data	Numeric matrix of reference data (n x d).
tol	Convergence tolerance for the adaptive stopping rule. Default 0.01.
batch_size	Number of random projections per batch. Default 100.
min_batches	Minimum batches before checking convergence. Default 5.
patience	Consecutive stable batches to declare convergence. Default 3.
seed	Integer random seed for reproducibility. Default 42.

Details

Projection depth is defined via the Stahel-Donoho outlyingness — the supremum over all directions of the robust univariate Z-score of the projected point, using median and MAD as location and scale. This makes it fully robust with a high breakdown point, and affine invariant.

The deepest point under projection depth is a genuine robust estimator of multivariate location.

Value

Numeric vector of depth values in (0, 1], one per query point.

References

Zuo, Y. & Serfling, R. (2000). General notions of statistical depth function. *Annals of Statistics*, 28(2), 461–482.

Examples

```
set.seed(42)
data <- matrix(rnorm(500), nrow = 100, ncol = 5)
x    <- matrix(rnorm(25),  nrow = 5,   ncol = 5)

projection_depth(x, data)

dd <- compute_depth(data, depth_fn = projection_depth)
```

```
median(dd)
outliers(dd)
```

`rank` *Rank*

Description

Generic function for ranking. For `depth` objects, returns depth-based ranks with rank 1 assigned to the deepest observation. For all other objects, delegates to `base::rank`.

Usage

```
rank(x, ...)
```

Arguments

`x`	An object. For `depth` objects, see `rank.depth`.
`...`	Additional arguments passed to methods.

Value

For `depth` objects, an integer vector of length n where rank 1 is the deepest observation. For other objects, see `rank`.

`rank.depth` *Depth-Based Ranks*

Description

Ranks observations by depth. Rank 1 is assigned to the deepest (most central) observation; rank n to the shallowest (most outlying).

Usage

```
## S3 method for class 'depth'
rank(x, ...)
```

Arguments

`x`	A `depth` object from `compute_depth()`.
`...`	Ignored.

Value

Integer vector of length n. Rank 1 = deepest.

`simplicial_depth` *Liu Simplicial Depth*

Description

Computes the simplicial depth of one or more query points with respect to a reference distribution estimated from `data`, using an adaptive Monte Carlo approximation with parallel computation.

Usage

```
simplicial_depth(
  x,
  data,
  tol = 0.05,
  batch_size = 200L,
  min_batches = 3L,
  max_batches = 20L,
  seed = 42L
)
```

Arguments

`x`	Numeric matrix of query points (m x d), or a numeric vector of length d for a single point.
`data`	Numeric matrix of reference data (n x d). Must have at least d+1 rows.
`tol`	Relative standard error tolerance for the adaptive stopping rule. Sampling stops when the standard error of the depth estimate drops below `tol` times the estimate itself. Default 0.05.
`batch_size`	Number of random simplices sampled per batch. Default 200.
`min_batches`	Minimum number of batches before checking convergence. Default 3.
`max_batches`	Maximum number of batches regardless of convergence. Acts as a hard cap on computation time. Default 20.
`seed`	Integer random seed for reproducibility. Default 42.

Details

Simplicial depth is the probability that a random simplex formed by d+1 points drawn from the data contains the query point. It is a genuine multivariate generalization of the median with strong geometric intuition and no distributional assumptions.

The deepest point — the simplicial median — is a robust estimator of location that reduces to the univariate median when d=1.

Value

Numeric vector of depth values in [0, 1], one per query point. Higher values indicate greater centrality.

References

Liu, R. Y. (1990). On a notion of data depth based on random simplices. *Annals of Statistics*, 18(1), 405–414.

Zuo, Y. & Serfling, R. (2000). General notions of statistical depth function. *Annals of Statistics*, 28(2), 461–482.

Examples

```
set.seed(42)
data <- matrix(rnorm(500), nrow = 100, ncol = 5)
x    <- matrix(rnorm(25),  nrow = 5,   ncol = 5)

# Basic usage
simplicial_depth(x, data)

# Via compute_depth for full depth object
dd <- compute_depth(data, depth_fn = simplicial_depth)
median(dd)
outliers(dd)
plot(dd)
```

---

spatial_depth *Spatial Depth*

Description

Computes the spatial depth of one or more query points with respect to a reference distribution estimated from `data`.

Usage

```
spatial_depth(x, data)
```

Arguments

`x`	Numeric matrix of query points (m x d), or a numeric vector of length d for a single point.
`data`	Numeric matrix of reference data (n x d).

Details

Spatial depth is defined as 1 minus the norm of the mean unit vector pointing from the data toward the query point. Unlike other depth functions in this package, it has a closed-form sample estimate with no Monte Carlo approximation required — making it the fastest depth function here, suitable for very large n and d.

Spatial depth is orthogonally invariant but not affine invariant. For affine invariant depth use `projection_depth` or `tukey_depth`.

Value

Numeric vector of depth values in [0, 1], one per query point.

References

Vardi, Y. & Zhang, C.-H. (2000). The multivariate L1-median and associated data depth. *Proceedings of the National Academy of Sciences*, 97(4), 1423–1426.

Examples

```
set.seed(42)
data <- matrix(rnorm(500), nrow = 100, ncol = 5)
x    <- matrix(rnorm(25),  nrow = 5,   ncol = 5)

spatial_depth(x, data)

dd <- compute_depth(data, depth_fn = spatial_depth)
median(dd)
outliers(dd)
```

---

tukey_depth *Tukey (Halfspace) Depth*

Description

Computes the Tukey halfspace depth of one or more query points with respect to a reference distribution estimated from `data`.

Usage

```
tukey_depth(
  x,
  data,
  tol = 0.01,
  batch_size = 100L,
  min_batches = 5L,
  patience = 3L,
  seed = 42L
)
```

Arguments

`x`	Numeric matrix of query points (m x d), or a numeric vector of length d for a single point.
`data`	Numeric matrix of reference data (n x d).

tol	Convergence tolerance for the adaptive stopping rule. Default 0.01 (1% relative change).
batch_size	Number of random projections per batch. Default 100.
min_batches	Minimum number of batches before checking convergence. Default 5.
patience	Number of consecutive stable batches to declare convergence. Default 3.
seed	Integer random seed for reproducibility. Default 42.

Details

Tukey depth is the canonical multivariate depth function. The deepest point — the Tukey median — is a genuine robust generalization of the univariate median, with breakdown point up to 1/(d+1). Depth is defined purely geometrically via halfspaces with no distributional assumptions.

Exact computation is O(n^(d-1)) and infeasible for d > 3. This implementation uses an adaptive random projection approximation: depth is estimated as the minimum over random unit vector projections of the fraction of data points on either side of the query point's projection. The stopping rule automatically determines when the estimate has stabilised.

Value

Numeric vector of depth values in [0, 0.5], one per query point.

References

Tukey, J. W. (1975). Mathematics and the picturing of data. *Proceedings of the International Congress of Mathematicians*, 2, 523–531.

Zuo, Y. & Serfling, R. (2000). General notions of statistical depth function. *Annals of Statistics*, 28(2), 461–482.

Examples

```
set.seed(42)
data <- matrix(rnorm(500), nrow = 100, ncol = 5)
x    <- matrix(rnorm(25),  nrow = 5,   ncol = 5)

# Basic usage
tukey_depth(x, data)

# Via compute_depth for full depth object
dd <- compute_depth(data, depth_fn = tukey_depth)
median(dd)
outliers(dd)
```

Index