Package 'EntropicStatistics'

November 5, 2026

Type Package

Title Functions Based on Entropic Statistics

Version 0.2.0

Description Nonparametric tools for comparing discrete distributions through generalized Shannon entropy profiles, following Zhang and Shi (2024) <doi:10.1016/j.spl.2024.110070>. HeatMap() visualizes these profiles across entropy orders. HeatMapOrder() provides hierarchical clustering, reference similarity ranking, and other row orderings, with accessible profiles and position tables. Comparisons are invariant to category permutations and describe distributional shape rather than dependence between variables.

License GPL-3

Encoding UTF-8

Depends R (>= 4.1.0)

Imports ggplot2 (>= 3.4.0), rlang, stats, utils

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

RoxygenNote 7.3.3

NeedsCompilation no

Author Jialin Zhang [aut, cph, cre],
Jingyi ``Catherine" Shi [aut] (Affiliation: Department of Mathematics and Statistics, Mississippi State University)

Maintainer Jialin Zhang <jzhang@math.msstate.edu>

Repository CRAN

Date/Publication 2026-11-04 23:30:11 UTC

Contents

HeatMap 2
HeatMapOrder 5

Index **9**

HeatMap *Heat Map of Generalized Shannon Entropy Profiles*

Description

Visualize discrete distributions on a common grid of entropy orders, with optional automatic row ordering and cluster boundaries.

Usage

```
HeatMap(
  data_frequency_list,
  orders = seq(0.50, 3, by = 0.01),
  selection = seq_along(data_frequency_list),
  plot_order = selection,
  RowNames = NULL,
  title = "HeatMap",
  x_ticks = NULL,
  plot_margin = ggplot2::margin(0.5, 0.2, 0.2, 1, "cm"),
  text_face = 1,
  fill_colors = c("blue4", "white", "red3"),
  title_text_size = 25,
  label_text_size = 25,
  sort = c("none", "cluster", "reference", "entropy", "name"),
  linkage = "average",
  reference = NULL,
  n_clusters = NULL,
  order_weights = NULL,
  top_n = NULL,
  show_legend = FALSE
)
```

Arguments

`data_frequency_list`
: A nonempty list of nonnegative, finite numeric frequency vectors, probability vectors, or one-dimensional tables. Each selected vector must contain at least one positive value. Category names need not agree across vectors. Names of list elements supply row labels.

`orders`
: A nonempty, strictly increasing vector of finite positive generalized Shannon entropy orders.

`selection`
: Distinct positive integer indices into `data_frequency_list`. Only these distributions are analyzed.

`plot_order`
: Original list indices in bottom-to-top display order. For manual ordering only (`sort = "none"`). If `selection` is also supplied, this must be a permutation of it; otherwise it also selects the rows.

RowNames	Optional nonmissing character labels. With manual ordering, these correspond to plot_order; with automatic ordering, to selection before sorting. Labels are moved with their distributions. Duplicate labels are supported. Missing list names receive labels such as "Group 2".
title	Plot title.
x_ticks	Orders to label on the x axis. Each must occur in orders (up to floating-point tolerance). The default selects up to five actual grid values.
plot_margin	A margin() object.
text_face	Font face: 1 = plain, 2 = bold, 3 = italic, 4 = bold italic, or the corresponding character value.
fill_colors	At least two colors, from low to high entropy.
title_text_size	Title text size.
label_text_size	Axis text size. Set to NULL for a size that decreases with the number of displayed rows. The historical default is retained.
sort	Row ordering: "none" preserves manual/input order; "cluster" groups similar profiles; "reference" places a chosen distribution first and then its nearest profiles; "entropy" sorts by weighted mean entropy; "name" sorts alphabetically. Automatic orders are read from top to bottom. The default remains "none" for compatibility.
linkage	Hierarchical linkage: "average", "complete", "single", or "ward.D2". Used only for cluster ordering.
reference	Original list index or unambiguous original list name of a selected distribution. Required for reference ordering. This refers to the input list, not to RowNames.
n_clusters	Optional number of groups for hierarchical clustering. Must lie between 1 and the number of selected rows. Adds cluster membership to the position table and horizontal boundaries to the plot. Only supported with cluster ordering; no number of clusters is inferred automatically.
order_weights	Optional nonnegative weights, one per order, with at least one positive entry. Normalized to sum to one. The default uses normalized trapezoidal weights over orders; see HeatMapOrder().
top_n	With reference ordering, optionally display only this many rows, including the reference. Other modes do not support this argument.
show_legend	Logical; show the entropy color scale (in nats).

Details

For probabilities p_k, the entropy profile is

$$H^{(m)} = -\sum_k q_k^{(m)} \ln q_k^{(m)}, \quad q_k^{(m)} = p_k^m / \sum_j p_j^m.$$

Natural logarithms are used. Zero counts are removed. Computation uses shifted log frequencies to avoid overflow and underflow in normalization.

Automatic ordering compares the full profiles on the supplied grid, without row standardization. It describes distributional shape up to permutation of category probabilities, not association between variables. A finite grid and empirical counts do not certify equality of distributions or identify distribution families. See `HeatMapOrder()` for the distance and ordering algorithm. Clustering is an exploratory extension to the heatmap in Zhang and Shi (2024), not an inferential procedure from that paper.

The x axis is a discrete grid: every supplied order occupies one tile. With an irregular grid, spacing on the plot does not represent numerical distance between orders. Ordering weights still account for grid spacing. Colors are scaled separately for each plot, as in earlier versions.

Value

A `ggplot` object. Its `"heatmap_order"` attribute is the list described in `HeatMapOrder()`, with display labels substituted when `RowNames` is supplied. For example, `attr(p, "heatmap_order")$table` lists rows from top to bottom.

References

Zhang, J. and Shi, J. (2024). Nonparametric clustering of discrete probability distributions with generalized Shannon's entropy and heatmap. *Statistics & Probability Letters*, 208, 110070. doi:10.1016/j.spl.2024.110070

See Also

`HeatMapOrder()`, `ggsave()`

Examples

```
set.seed(1)
binom_n <- 10
sample_size <- 400
sample_1 <- table(stats::rbinom(n = sample_size, size = binom_n, prob = 0.1))
sample_2 <- table(stats::rbinom(n = sample_size, size = binom_n, prob = 0.2))
sample_3 <- table(stats::rbinom(n = sample_size, size = binom_n, prob = 0.3))
sample_4 <- table(stats::rbinom(n = sample_size, size = binom_n, prob = 0.4))
sample_5 <- table(stats::rbinom(n = sample_size, size = binom_n, prob = 0.5))
sample_6 <- table(stats::rbinom(n = sample_size, size = binom_n, prob = 0.6))
sample_7 <- table(stats::rbinom(n = sample_size, size = binom_n, prob = 0.7))
sample_8 <- table(stats::rbinom(n = sample_size, size = binom_n, prob = 0.8))
sample_9 <- table(stats::rbinom(n = sample_size, size = binom_n, prob = 0.9))
poisson_1 <- table(stats::rpois(n = sample_size, lambda = 1))
poisson_2 <- table(stats::rpois(n = sample_size, lambda = 2))
poisson_3 <- table(stats::rpois(n = sample_size, lambda = 3))
poisson_4 <- table(stats::rpois(n = sample_size, lambda = 4))
poisson_5 <- table(stats::rpois(n = sample_size, lambda = 5))
poisson_6 <- table(stats::rpois(n = sample_size, lambda = 6))
poisson_7 <- table(stats::rpois(n = sample_size, lambda = 7))
poisson_8 <- table(stats::rpois(n = sample_size, lambda = 8))
poisson_9 <- table(stats::rpois(n = sample_size, lambda = 9))
data_samples <- list(
  binom_0.1 = sample_1, binom_0.2 = sample_2, binom_0.3 = sample_3,
```

```
  binom_0.4 = sample_4, binom_0.5 = sample_5, binom_0.6 = sample_6,
  binom_0.7 = sample_7, binom_0.8 = sample_8, binom_0.9 = sample_9,
  Poisson_1 = poisson_1, Poisson_2 = poisson_2, Poisson_3 = poisson_3,
  Poisson_4 = poisson_4, Poisson_5 = poisson_5, Poisson_6 = poisson_6,
  Poisson_7 = poisson_7, Poisson_8 = poisson_8, Poisson_9 = poisson_9
)
HeatMap(data_samples)
HeatMap(data_samples, sort = "cluster")
HeatMap(data_samples, selection = sample(seq_along(data_samples), 6))
HeatMap(data_samples, selection = 1:9)
HeatMap(data_samples, selection = 10:13)
HeatMap(data_samples, selection = 14:18)


# Small examples illustrating the new ordering options.
samples <- list(A = c(70, 20, 10), B = c(10, 70, 20),
                C = c(33, 34, 33), D = c(95, 4, 1))
HeatMap(samples, label_text_size = 11)
p <- HeatMap(samples, sort = "cluster", n_clusters = 2,
             label_text_size = NULL, show_legend = TRUE)
p
attr(p, "heatmap_order")$table
HeatMap(samples, sort = "reference", reference = "A", top_n = 3,
        label_text_size = 11)
# Old manual ordering remains available (bottom to top).
HeatMap(samples, plot_order = c(3, 1, 2), label_text_size = 11)
```

`HeatMapOrder` *Order and Inspect Generalized Shannon Entropy Profiles*

Description

Compute entropy profiles, arrange similar distributions together, and return a searchable table of row positions without drawing a plot.

Usage

```
HeatMapOrder(
  data_frequency_list,
  orders = seq(0.50, 3, by = 0.01),
  selection = seq_along(data_frequency_list),
  method = c("cluster", "reference", "entropy", "name", "none"),
  linkage = "average",
  reference = NULL,
  n_clusters = NULL,
  order_weights = NULL,
  top_n = NULL
)
```

Arguments

`data_frequency_list`
: A nonempty list of nonnegative, finite numeric frequency vectors, probability vectors, or one-dimensional tables. Each selected vector must contain at least one positive value. Category names need not agree across vectors. Names of list elements supply row labels.

`orders`
: A nonempty, strictly increasing vector of finite positive generalized Shannon entropy orders.

`selection`
: Distinct positive integer indices into `data_frequency_list`. Only these distributions are analyzed.

`method`
: Ordering method. `"cluster"` is the default for this function; the alternatives are `"reference"`, `"entropy"`, `"name"`, and `"none"`.

`linkage`
: Hierarchical linkage: `"average"`, `"complete"`, `"single"`, or `"ward.D2"`. Used only for cluster ordering.

`reference`
: Original list index or unambiguous original list name of a selected distribution. Required for reference ordering. This refers to the input list, not to `RowNames`.

`n_clusters`
: Optional number of groups for hierarchical clustering. Must lie between 1 and the number of selected rows. Adds cluster membership to the position table and horizontal boundaries to the plot. Only supported with cluster ordering; no number of clusters is inferred automatically.

`order_weights`
: Optional nonnegative weights, one per order, with at least one positive entry. Normalized to sum to one. The default uses normalized trapezoidal weights over `orders`; see `HeatMapOrder()`.

`top_n`
: With reference ordering, optionally display only this many rows, including the reference. Other modes do not support this argument.

Details

The distance between profiles H_i and H_j is

$$d(i,j) = \sqrt{\sum_l w_l (H_i(m_l) - H_j(m_l))^2}.$$

For two or more orders, default weights are normalized trapezoidal integration weights over the supplied grid. This is a discrete approximation to the root mean squared difference over the order interval, so an unevenly sampled part of that interval is not weighted merely by its number of points. With a single order, its weight is one. User weights replace these defaults; using `rep(1, length(orders))` weights every column equally.

`"cluster"` applies `hclust()` to Euclidean distances between profiles multiplied by the square roots of their weights. By default, average linkage is used. Branches are rotated with `reorder()` and weighted mean entropies (`agglo.FUN = mean`) to make the display easier to scan, without changing cluster membership. This is a deterministic heuristic for fixed inputs, not optimal leaf ordering. Exact ties can depend on input order. Constant/uniform profiles are supported.

`"reference"` puts the reference first, then the other rows in increasing distance from it, breaking ties by selection order. It computes only distances to the reference. `top_n` limits the displayed

result, not the analyzed input. "entropy" sorts weighted mean entropies increasingly; "name" uses case-sensitive radix ordering of row names. "none" preserves selection as the bottom-to-top plotting order.

Distances are computed on raw entropies in nats. Row standardization would discard entropy levels and is deliberately not performed. Category labels are ignored, so permutations of the same frequency vector have identical profiles. This comparison concerns marginal distributions, not dependence or correlation between variables. A finite profile grid is descriptive and does not establish distributional equality; sampling uncertainty and the choice of orders or bins can affect the result.

Hierarchical clustering requires quadratic memory in the number of rows. For very large lists, prefer reference or name ordering, or select a subset.

Value

A list with:

plot_order Original list indices, bottom to top; can be passed directly to HeatMap(..., plot_order = result$plot_order).

table A data frame in top-to-bottom display order. position is the displayed position, index is the original list index, label is the row label, cluster is membership when n_clusters was supplied (otherwise NA), mean_entropy is weighted mean entropy, and distance is distance to the reference (otherwise NA). Cluster numbers follow their first top-to-bottom appearance.

profiles Numeric matrix in selection order, with one column per entropy order. Retains all selected rows even when top_n is used.

selection Original list indices of all analyzed rows.

orders, order_weights The order grid and normalized weights.

method The ordering method used.

distance A dist object for cluster ordering with at least two rows, otherwise NULL.

clustering The hclust object with leaf order matching the table (positions within selection), or NULL if no tree was constructed.

References

Zhang, J. and Shi, J. (2024). Nonparametric clustering of discrete probability distributions with generalized Shannon's entropy and heatmap. *Statistics & Probability Letters*, 208, 110070. doi:10.1016/j.spl.2024.110070

See Also

HeatMap(), hclust(), cutree()

Examples

```
samples <- list(A = c(70, 20, 10), B = c(10, 70, 20),
                C = c(33, 34, 33), D = c(95, 4, 1))
result <- HeatMapOrder(samples, n_clusters = 2)
result$table
result$profiles
```

```
HeatMap(samples, plot_order = result$plot_order, label_text_size = 11)
nearest <- HeatMapOrder(samples, method = "reference", reference = "A")
nearest$table
```

Index

∗ **cluster**
HeatMapOrder, 5
∗ **hplot**
HeatMap, 2

cutree, *7*

ggsave, *4*

hclust, *6, 7*
HeatMap, 2, *7*
HeatMapOrder, *3, 4*, 5, *6*

margin, *3*

reorder, *6*