Package 'nomiShape'

May 9, 2026

Title Visualization and Analysis of Nominal Variable Distributions

Version 1.0.2

Description Provides tools for visualizing and analyzing the shape of discrete nominal frequency distributions. The package introduces centered frequency plots, in which nominal categories are ordered from the most frequent category at the center toward less frequent categories on both sides, facilitating the detection of distributional patterns such as uniformity, dominance, symmetry, skewness, and long-tail behavior. In addition, the package supports Pareto charts for the study of dominance and cumulative frequency structure in nominal data. The package is designed for exploratory data analysis and statistical teaching, offering visualizations that emphasize distributional form rather than arbitrary category ordering.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 4.1.0)

LazyData true

Imports dplyr, ggplot2

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation no

Author Norberto Asensio [aut, cre] (ORCID:
<https://orcid.org/0000-0003-4536-5073>)

Maintainer Norberto Asensio <norberto.asensio@ehu.eus>

Repository CRAN

Date/Publication 2026-03-21 18:40:02 UTC

Contents

`alice` *Alice in Wonderland word dataset*

Description

Tokenized words from *Alice's Adventures in Wonderland* by Lewis Carroll. Each row represents a single word occurrence, allowing analysis of word frequency distributions and demonstration of Zipf-like behavior in natural language.

Usage

```
alice
```

Format

A data frame with one column:

word Character. Tokenized word from the text.

Source

Public domain text (Lewis Carroll, 1865)

`categories` *Categories: Uniform Distribution of Bikinibottom Species*

Description

A dataset of dummy nominal data inspired by characters/species from the Bikini Bottom universe (SpongeBob SquarePants). This dataset simulates a roughly uniform distribution across 11 species, with a total of 250 observations. It was intentionally designed to be uniform-like for testing nominal distribution visualization functions.

A simple dataset of categorical values used for examples.

Usage

```
categories

categories
```

Format

A data frame with 250 rows and 1 variable:

animal Character. Species/animal names. 11 species inspired by Bikini Bottom.

A data frame with 1 column:

animal Factor with animal categories as letters

Source

Generated for examples

Examples

```
categories
# Ranked bar plot of species frequencies
ranked_barplot(categories, "animal")

# Centered bar plot (most frequent in the center)
centered_barplot(categories, "animal")

# Centered dot plot with theoretical shape overlays
shape_comp_plot(categories, "animal")
```

categories2 *Categories2: Triangular Distribution of Bikinibottom Species*

Description

A dataset of dummy nominal data inspired by characters/species from the Bikini Bottom universe (SpongeBob SquarePants). This dataset simulates a roughly triangular distribution of frequencies.

Usage

```
categories2
```

Format

A data frame with 250 rows and 11 variables:

animal Character. Species/animal names.

freq Integer. Frequency of each species, forming a triangular pattern.

Examples

```
ranked_barplot(categories2, "animal")
```

categories3 *Categories3: Exponential/Dominance Distribution of Bikinibottom Species*

Description

A dataset of dummy nominal data inspired by characters/species from the Bikini Bottom universe (SpongeBob SquarePants). This dataset simulates a highly skewed distribution where a few species dominate most of the frequency (long-tail / exponential pattern). It was intentionally designed for pedagogical purposes to demonstrate dominance and Pareto-like behavior in nominal data.

Usage

```
categories3
```

Format

A data frame with 250 rows and 1 variable:

animal Character. Species/animal names. 11 species inspired by Bikini Bottom.

Examples

```
categories3
# Centered dot plot showing exponential/long-tail pattern
shape_comp_plot(categories3, "animal")

# Pareto chart highlighting cumulative frequency and dominance
pareto(categories3, "animal")

# Optional: ranked or centered bar plots
ranked_barplot(categories3, "animal")
centered_barplot(categories3, "animal")
```

`categories4` *Categories4: Structured (triangular / normal-like) nominal distribution*

Description

A dataset of dummy nominal data inspired by characters or species from the Bikini Bottom universe (SpongeBob SquarePants).

Usage

```
categories4
```

Format

A data frame with 250 rows and 1 variable:

animal Character or species name (11 categories inspired by Bikini Bottom).

Details

The dataset represents a structured nominal distribution in which a limited number of categories dominate, followed by a gradual and approximately symmetric decline in frequencies. This pattern is consistent with a triangular or normal-like shape rather than a strongly long-tailed (Pareto/exponential) distribution.

The dataset was intentionally designed for pedagogical purposes to illustrate dominance, symmetry, and modal structure in nominal data, and to serve as a contrast with truly long-tailed distributions included elsewhere in the package.

Examples

```
categories4

# Centered dot plot showing a structured (normal-like) pattern
shape_comp_plot(categories4, "animal")

# Pareto chart showing dominance without a strong long tail
```

```
pareto(categories4, "animal")

# Ranked and centered bar plots
ranked_barplot(categories4, "animal")
centered_barplot(categories4, "animal")
```

`centered_barplot` *Centered Frequency Bar Plot for Nominal Variables Creates a centered bar plot for discrete nominal variables by placing the most frequent category at the center and progressively less frequent categories alternately to the left and right.*

Description

Centered Frequency Bar Plot for Nominal Variables Creates a centered bar plot for discrete nominal variables by placing the most frequent category at the center and progressively less frequent categories alternately to the left and right.

Usage

```
centered_barplot(df, var, title = NULL, scale = c("count", "percent"))
```

Arguments

`df`	A data frame containing the nominal variable.
`var`	A character string giving the name of the nominal variable in `df`.
`title`	Optional character string specifying the plot title.
`scale`	Character string specifying the scale of the frequencies: `"count"` (default) for raw counts or `"percent"` for percentages.

Value

A `ggplot2` object.

Examples

```
centered_barplot(categories, "animal")
centered_barplot(categories, "animal", scale = "percent")
```

`centered_dotplot` *Centered Dot Plot for Nominal Variables*

Description

Creates a centered dot plot for a nominal variable, ordering categories from the most frequent at the center toward less frequent categories on both sides. Optionally connects points with a line and shades the area under the line.

Usage

```
centered_dotplot(
  df,
  var,
  connect = FALSE,
  shade = FALSE,
  scale = c("count", "percent")
)
```

Arguments

`df`	A data.frame or tibble containing the variable.
`var`	Character. Name of the nominal variable in `df`.
`connect`	Logical; if TRUE, connects points with a line.
`shade`	Logical; if TRUE, shades the area under the line (requires connect = TRUE).
`scale`	Character; either `"count"` (default) or `"percent"`.

Value

A `ggplot2` object.

Examples

```
centered_dotplot(categories, "animal")
centered_dotplot(categories, "animal", connect = TRUE)
centered_dotplot(categories, "animal", connect = TRUE, shade = TRUE)
centered_dotplot(mpg, "manufacturer", scale = "percent")
```

central_concentration *Central Concentration Index for Nominal Variables*

Description

Computes a measure of how concentrated counts are around the center of a nominal variable, based on the centered plotting order.

Usage

```
central_concentration(df, var, top_k = 3, weighted = FALSE)
```

Arguments

df	A data.frame or tibble containing the variable.
var	Character. Name of the nominal variable in df.
top_k	Numeric. Number of central categories to consider (default: 3).
weighted	Logical. If TRUE, applies a weight decreasing with distance from center.

Value

A numeric value between 0 and 1 representing the central concentration.

Examples

```
central_concentration(categories, "animal")
central_concentration(categories2, "animal", top_k = 5)
central_concentration(categories3, "animal", weighted = TRUE)
```

dominance_index *Dominance Index for Nominal Variables*

Description

Computes dominance for a nominal variable using the Simpson index, quantifying the degree to which a few categories dominate the distribution.

Usage

```
dominance_index(df, var)
```

Arguments

df	A data.frame or tibble containing the nominal variable.
var	Character. Name of the nominal variable in df.

Details

Dominance is calculated as:

$$D = \sum p_i^2$$

where p_i is the relative frequency of category i.

Higher values indicate stronger dominance by fewer categories.

Value

A numeric value representing dominance.

Examples

```
dominance_index(categories, "animal")
dominance_index(categories2, "animal")
dominance_index(categories3, "animal")
```

---

kafka *The Metamorphosis word dataset*

Description

Tokenized words from *The Metamorphosis* by Franz Kafka. Each row represents a single word occurrence, allowing analysis of word frequency distributions and comparison with Zipf's law.

Usage

```
kafka
```

Format

A data frame with one column:

word Character. Tokenized word from the text.

Source

Public domain text (Franz Kafka, 1915; English translation)

mpg *MPG dataset*

Description

Car fuel economy data (from ggplot2) for examples.

Usage

```
mpg
```

Format

A data frame

Source

ggplot2::mpg

pareto *Pareto Plot for Nominal Variables*

Description

Creates a Pareto chart for a nominal variable, displaying frequencies and cumulative percentages.

Usage

```
pareto(df, var, show_table = TRUE)
```

Arguments

df	A data.frame or tibble containing the variable.
var	Character. Name of the variable in df.
show_table	Logical; if TRUE, prints the frequency table. Default is FALSE.

Value

A ggplot2 object representing the Pareto chart.

Examples

```
pareto(categories, "animal")
```

pielou_evenness *Pielou's Evenness for Nominal Variables*

Description

Computes Pielou's evenness index based on Shannon entropy for a nominal variable recorded as individual-level observations.

Usage

```
pielou_evenness(df, var)
```

Arguments

df	A data.frame or tibble containing the nominal variable.
var	Character string giving the name of the nominal variable in df.

Details

Pielou's evenness is defined as:

$$E = H/\log(S)$$

where H is Shannon entropy and S is the number of observed categories.

Values range from 0 (complete dominance by one category) to 1 (perfectly even distribution).

Value

A numeric value representing Pielou's evenness.

Examples

```
pielou_evenness(categories, "animal")
pielou_evenness(categories2, "animal")
pielou_evenness(categories3, "animal")
```

ranked_barplot *Ranked Bar Plot for Nominal Variables*

Description

Creates a bar plot for a nominal variable, with categories ordered from most frequent to least frequent.

Usage

```
ranked_barplot(df, var, scale = c("count", "percent"), title = NULL)
```

Arguments

df	A data.frame or tibble containing the variable.
var	Character string giving the name of the variable in df.
scale	Character; either "count" (default) or "percent".
title	Optional character string specifying the plot title.

Value

A ggplot2 object representing the ranked bar plot.

Examples

```
ranked_barplot(categories, "animal")
ranked_barplot(categories, "animal", scale = "percent")
```

---

ranked_dotplot *Ranked Dot Plot for Nominal Variables*

Description

Creates a ranked dot plot for a nominal variable, displaying category frequencies or percentages from highest to lowest. Optionally connects points with a line and shades the area under the line.

Usage

```
ranked_dotplot(
  df,
  var,
  connect = FALSE,
  shade = FALSE,
  scale = c("count", "percent")
)
```

Arguments

df	A data.frame or tibble containing the variable.
var	Character. Name of the nominal variable in df.
connect	Logical; if TRUE, connects points with a line.
shade	Logical; if TRUE, shades the area under the line. Default is FALSE.
scale	Character; either "count" (default) or "percent".

Value

A ggplot2 object.

Examples

```
ranked_dotplot(categories, "animal")
ranked_dotplot(categories, "animal", connect = TRUE)
ranked_dotplot(categories, "animal", connect = TRUE, shade = TRUE)
ranked_dotplot(mpg, "manufacturer", scale = "percent")
```

---

rare_plot *Rarefaction curve for nominal variables*

Description

Generates a rarefaction curve showing the expected number of distinct categories discovered as sampling effort increases. The curve is estimated using Monte Carlo permutations of the observation order.

Usage

```
rare_plot(df, var, reps = 1000, max_effort = NULL)
```

Arguments

df	A data frame containing the nominal variable.
var	Character string specifying the nominal variable column.
reps	Number of random permutations used to estimate the curve. The default is 1000. Smaller values can be used to reduce computation time when working with large datasets, at the cost of less precise confidence intervals.
max_effort	Maximum sampling effort to compute. If NULL (default), the full sample size is used. For very large datasets, this argument allows users to limit the rarefaction curve to a smaller number of observations in order to explore how quickly categories accumulate and to approximate the minimum sample size required to capture most of the category diversity.

Value

Invisibly returns a data frame containing:

- effort: sampling effort
- mean: expected number of categories
- lowCI: lower confidence interval
- highCI: upper confidence interval

Examples

```
rare_plot(categories3, "animal")
rare_plot(ufo, "shape", reps = 25, max_effort = 500)
```

shape_aic *Fit Nominal Data to Theoretical Shapes Using AIC (Safe Exponential)*

Description

Computes the multinomial log-likelihood of observed counts against four theoretical distributions (uniform, triangular, normal-like, and exponential/Pareto-like) and returns AIC and DeltaAIC values.

Usage

```
shape_aic(df, var, rate_exp = 0.7, eps = 1e-12)
```

Arguments

df	A data.frame or tibble containing the nominal variable.
var	Character string giving the name of the nominal variable in df.
rate_exp	Numeric. Default exponential rate. Only used if tail not clearly exponential.
eps	Small numeric value added to probabilities to avoid log(0). Default is 1e-12.

Value

A data.frame with columns: Shape, AIC, DeltaAIC.

shape_comp_plot *Compare Observed Nominal Distribution with Theoretical Shapes*

Description

Plots a centered dotplot of a nominal variable and overlays four theoretical distributions: uniform, triangular, exponential (Pareto-like), and normal-like.

Usage

```
shape_comp_plot(df, var, rate_exp = 0.7, scale = c("count", "percent"))
```

Arguments

df	A data.frame or tibble containing the nominal variable.
var	Character string giving the name of the nominal variable in df.
rate_exp	Numeric. Rate parameter for the exponential distribution (Pareto-like). Default is 0.7.
scale	Character. Whether to scale frequencies as counts ("count") or percentages ("percent"). Default is "count".

Details

The function orders categories from most frequent at the center outwards. Observed frequencies are plotted as points and lines, and each theoretical distribution is overlaid with a different color and line type.

Value

A ggplot2 object.

Examples

```
shape_comp_plot(categories, "animal")
shape_comp_plot(categories2, "animal")
shape_comp_plot(categories3, "animal")
```

---

starwars *Star Wars dataset*

Description

Character info from Star Wars (from dplyr/ggplot2 examples)

Usage

```
starwars
```

Format

A data frame

Source

dplyr::starwars

tail_index *Tail Index for Nominal Variables*

Description

Computes the proportion of categories contributing to the lower part of the distribution. Useful to quantify long-tail structure in nominal distributions.

Usage

```
tail_index(df, var, threshold = 0.8)
```

Arguments

df	A data.frame or tibble containing the variable.
var	Character. Name of the nominal variable in df.
threshold	Numeric. Cumulative proportion of counts defining the "dominant" categories (default 0.8).

Value

Numeric between 0 and 1 representing the tail proportion.

Examples

```
tail_index(categories3, "animal")
tail_index(categories2, "animal", threshold = 0.9)
```

ufo *UFO Sightings Dataset*

Description

Sample data of UFO sightings used for examples.

A large real-world dataset of UFO sighting reports collected by the National UFO Reporting Center (NUFORC), a non-profit organization dedicated to the collection and dissemination of objective UFO data.

Usage

```
ufo
```

```
ufo
```

Format

A data frame with 8 columns:

city Character. City where the UFO was reported.

comments Character. Description or comments about the sighting.

date_sighted Date. Date of the sighting.

duration_sec Numeric. Duration of the sighting in seconds.

latitude Numeric. Latitude of the sighting location.

longitude Numeric. Longitude of the sighting location.

shape Factor. Shape of the UFO observed. This is the nominal variable of interest.

state Character. State of the sighting location.

A data frame with 63,561 rows and 8 variables:

date_sighted Character. Date of the UFO sighting (YYYY-MM-DD).

latitude Numeric. Latitude of the sighting location.

longitude Numeric. Longitude of the sighting location.

city Character. City where the sighting occurred.

state Character. State or region of the sighting.

shape Character. Reported shape of the UFO (nominal variable of interest).

duration_sec Numeric. Duration of the sighting in seconds.

comments Character. Free-text comments describing the sighting.

Details

The dataset contains over 63,000 reported sightings spanning several decades and includes information on sighting date, geographic location, duration, narrative comments, and—most importantly for nomiShape— the reported *shape* of the observed object.

The shape variable is a nominal variable with many categories (e.g., "light", "circle", "triangle", "sphere"), exhibiting strong dominance by a few common shapes followed by a gradual decline across rarer categories. Despite the presence of a highly frequent leading category ("light"), the overall frequency structure is better described as triangular or normal-like rather than strictly exponential or Pareto.

This dataset is included as a realistic, large-sample example for exploring dominance, modality, and shape classification of nominal distributions using visual and information-theoretic tools.

Source

Example dataset

National UFO Reporting Center (NUFORC), https://nuforc.org

Examples

```
ufo

# Centered bar plot highlighting dominance and symmetry
centered_barplot(ufo, "shape")

# Centered dot plot with connections and shading
centered_dotplot(ufo, "shape", connect = TRUE, shade = TRUE)

# Shape comparison plot
shape_comp_plot(ufo, "shape")

# AIC-based shape classification
shape_aic(ufo, "shape")
```

---

zipf_rank_plot *Rank-frequency (Zipf) plot for nominal variables*

Description

Generates a rank-frequency plot comparing observed category frequencies with the expected Zipf distribution (inverse rank relationship).

Usage

```
zipf_rank_plot(df, var, max_rank = NULL, top_prop = NULL, loglog = FALSE)
```

Arguments

df	A data frame containing the nominal variable.
var	Character string specifying the nominal variable column.
max_rank	Maximum number of ranks to display. If NULL (default), all ranks are shown.
top_prop	Proportion of total observations to retain (0–1). If set, only the most frequent categories accounting for this cumulative proportion are displayed. Overrides max_rank.
loglog	Logical. If TRUE, both axes are displayed on a log10 scale.

Value

Invisibly returns a data frame with rank-frequency information.

Examples

```
zipf_rank_plot(kafka, "word")
zipf_rank_plot(alice, "word", loglog=TRUE)
zipf_rank_plot(alice, "word", max_rank = 250)
```

Index