

# Package ‘Logolas’

April 12, 2018

**Type** Package

**Title** Flexible and Customized Logo Plots using symbols, alphabets, numbers and alphanumeric strings

**Version** 1.2.0

**Date** 11.5.2016.

**Maintainer** Kushal Dey <kkdey@uchicago.edu>

**Description** Produces logo plots of a variety of symbols and names comprising English alphabets, numerics and punctuations. Can be used for sequence motif generation, mutation pattern generation, protein amino acid generation and symbol strength representation in any generic context.

**License** GPL (>= 2)

**LazyData** TRUE

**Depends** R (>= 3.4)

**URL** <https://github.com/kkdey/Logolas>

**Imports** grid, graphics

**Suggests** knitr, BiocStyle, Biobase, roxygen2, devtools, xtable, gridExtra, RColorBrewer, seqLogo, aRxiv

**VignetteBuilder** knitr

**biocViews** SequenceMatching, Alignment, Software, Visualization

**RoxygenNote** 5.0.1

**NeedsCompilation** no

**Author** Kushal Dey [aut, cre]

## R topics documented:

himalayan_fauna_2_clusters . . . . .	2
himalayan_fauna_3_clusters . . . . .	2
ic_computer . . . . .	3
logomaker . . . . .	4
makemylogo . . . . .	6
<b>Index</b>	<b>8</b>

---

himalayan\_fauna\_2\_clusters

*Phylogenetic abundance of bird species families in two clusters of regions in Himalayas*

---

**Description**

Phylogenetic abundance of bird species families in two clusters of regions in Himalayas

**Usage**

himalayan\_fauna\_2\_clusters

**Format**

A matrix with bird species families along rows and the clusters along columns

**Value**

A matrix with bird species families along rows and the clusters along columns

---

himalayan\_fauna\_3\_clusters

*Phylogenetic abundance of bird species families in three clusters of regions in Himalayas*

---

**Description**

Phylogenetic abundance of bird species families in three clusters of regions in Himalayas

**Usage**

himalayan\_fauna\_3\_clusters

**Format**

A matrix with bird species families along rows and the clusters along columns

**Value**

A matrix with bird species families along rows and the clusters along columns

---

ic_computer	<i>Information criterion computer</i>
-------------	---------------------------------------

---

## Description

Computes information criterion based on Renyi entropy for determining the size of the logos in the logo plot. Similar but allows for more general entropy criteria to the `pwm2ic()` function in `seqLogo` package. By tuning the `alpha` parameter input, one can vary the entropy from Shannon (`alpha=1`) to collision (`alpha=2`) to min-entropy (`alpha=infty`)

## Usage

```
ic_computer(mat, alpha, hist = FALSE)
```

## Arguments

<code>mat</code>	A matrix with symbols or logo names along the rows and the sites/positions/groups along the columns.
<code>alpha</code>	The tuning parameter of the Renyi entropy used in computing the information criterion. Default is <code>alpha=1</code> , for which it uses Shannon entropy (in the limit).
<code>hist</code>	if <code>hist</code> is <code>FALSE</code> (default), information criterion is used to decide on the heights of the logo plots. If <code>TRUE</code> , one uses the relative proportion of the values in the different columns of the matrix to determine the height of the bars.

## Value

A vector of same length as the number of columns in the data, with each entry representing information contained in that column, which determines the height of the bar for the logo plot for that particular column (site/position/block).

## Examples

```
counts_mat <- rbind(c(0, 10, 100, 60, 20),
                  c(40, 30, 30, 35, 20),
                  c(100, 0, 15, 25, 75),
                  c(10, 30, 20, 50, 70)
                  )
colnames(counts_mat) <- c("2012", "2013", "2014", "2015", "2016")
rownames(counts_mat) <- c("P1", "P2", "P3", "P4")
ic_computer(counts_mat, alpha=2)
```

---

logomaker

*Main workhorse function that builds the logo plots*


---

## Description

stacks logos created by the makemylogo function on top of each other to build the logo plot.

## Usage

```
logomaker(table, ic = NULL, hist = FALSE, color_profile,
  total_chars = c("A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K", "L",
    "M", "N", "O", "P", "Q", "R", "S", "T", "U", "V", "W", "X", "Y", "Z", "zero",
    "one", "two", "three", "four", "five", "six", "seven", "eight", "nine", "dot",
    "comma", "dash", "colon", "semicolon", "leftarrow", "rightarrow"),
  frame_width = NULL, ic.scale = TRUE, alpha = 1, xaxis = TRUE,
  yaxis = TRUE, xaxis_fontsize = 10, xlab_fontsize = 15,
  y_fontsize = 15, main_fontsize = 16, start = 0.001,
  yscale_change = TRUE, pop_name = NULL, xlab = "X",
  ylab = "Information content", col_line_split = "grey80", scale0 = 0.01,
  scale1 = 0.99, addlogos = NULL, addlogos_text = NULL, newpage = TRUE)
```

## Arguments

table	The input table (data frame or matrix) of counts across different logos or symbols (specified along the rows) and across different sites or positions or groups (specified along the columns).
ic	A vector of same length as the number of columns in the table, representing the heights of the logo stacked bars for each position/site/block. It defaults to NULL, in which case, the function computes the ic vector using the ic_computer functionality.
hist	Whether to use the hist method or the information criterion method to determine the heights of the logos.
color_profile	A list containing two elements - "type" and "col". The type can be of three types - "per-row", "per-column" and "per-symbol". The "col" element is a vector of colors, of same length as number of rows in table for "per-row" (assigning a color to each string), of same length as number of columns in table for "per-column" (assuming a color for each column), or a distinct color for a distinct symbol in "per-symbol". For "per-symbol", the length of the color_profile\$col should be same as library size of the logos, but if the vector of colors provided is more or less, we can downsample or upsample the colors as required. The colors are matched with the symbols in the total_chars
total_chars	The total number of character symbols in the user library. The default is the default library provided by Logolas, but the user can add symbols that he creates to this list.
frame_width	The width of the frames for individual site/position/column in the logo plot. As default, all the columns have same width, equal to 1.
ic.scale	if TRUE, the height of the bars in the stacked logo chart for each column is determined based on the information criterion input. Otherwise, the bars are normalized so that the height of each bar is 1\$. Defaults to TRUE.

<code>alpha</code>	The Renyi entropy tuning parameter which is used in case of scaling of the bar heights by information criterion. The default tuning parameter value is 1, which corresponds to Shannon entropy.
<code>xaxis</code>	Binary specifying if there should be a X axis in the logo plot or not. Defaults to TRUE.
<code>yaxis</code>	Binary specifying if there should be a Y axis in the logo plot or not. Defaults to TRUE.
<code>xaxis_fontsize</code>	The size of the X-axis axis ticks.
<code>xlab_fontsize</code>	The size of the X-axis label.
<code>y_fontsize</code>	The size of the Y-axis font.
<code>main_fontsize</code>	The size of the title.
<code>start</code>	The starting point in Y axis for the first logo. Default is 0.0001 which is very close to 0.
<code>yscale_change</code>	If TRUE, adjusts the Y axis scale based on the size of the bars, else keeps it to the maximum value possible, which is <code>ceiling(max(ic))</code> under <code>ic_computer</code> defined IC criteria.
<code>pop_name</code>	User can mention a name of the population for which the logo plot is created. Defaults to NULL when no population name is mentioned.
<code>xlab</code>	X axis label
<code>ylab</code>	Y axis label
<code>col_line_split</code>	The color of the line split between the consecutive groups or blocks
<code>scale0</code>	the base change of the logo to maintain the gap between symbols.
<code>scale1</code>	scaling of the logo to maintain the gap between symbols.
<code>addlogos</code>	Vector of additional logos/symbols defined by user
<code>addlogos_text</code>	Vector of the names given to the additional logos/symbols defined by user.
<code>newpage</code>	if TRUE, plots the logo plot in a new page. Defaults to TRUE.

### Value

Plots the logo plot for the table data, with column names representing the sites/blocks and the row names denoting the symbols for which logos are plotted

### Examples

```
cols = RColorBrewer::brewer.pal.info[RColorBrewer::brewer.pal.info$category == 'qual',]
col_vector = unlist(mapply(RColorBrewer::brewer.pal, cols$maxcolors, rownames(cols)))
counts_mat <- rbind(c(0, 10, 100, 60, 20),
                   c(40, 30, 30, 35, 20),
                   c(100, 0, 15, 25, 75),
                   c(10, 30, 20, 50, 70))
colnames(counts_mat) <- c("2012", "2013", "2014", "2015", "2016")
rownames(counts_mat) <- c("MAN", "MAIL", "LAWN", "CAR")

color_profile <- list("type" = "per_symbol",
                    "col" = col_vector)
logomaker(counts_mat,
          color_profile = color_profile,
          frame_width = 1,
```

```

        ic.scale = FALSE)

color_profile <- list("type" = "per_row",
                    "col" = col_vector[1:4])

logomaker(counts_mat,
          color_profile = color_profile,
          frame_width = 1,
          ic.scale = FALSE)

color_profile <- list("type" = "per_column",
                    "col" = col_vector[1:5])

logomaker(counts_mat,
          color_profile = color_profile,
          frame_width = 1,
          ic.scale = FALSE)

```

---

makemylogo

*Logo maker for a given English alphanumeric with common punctuations*


---

## Description

Plots logo for a given english symbol or name that contains English alphabets, numbers or punctuations like dots, dashes, etc. This is the skeleton used by the `logomaker` function of the package to create distinct logos for distinct alphanumeric symbols.

## Usage

```

makemylogo(name, colfill = "orange", plot = FALSE, total_chars = c("A",
  "B", "C", "D", "E", "F", "G", "H", "I", "J", "K", "L", "M", "N", "O", "P",
  "Q", "R", "S", "T", "U", "V", "W", "X", "Y", "Z", "zero", "one", "two",
  "three", "four", "five", "six", "seven", "eight", "nine", "dot", "comma",
  "dash", "colon", "semicolon", "leftarrow", "rightarrow"), addlogos = NULL,
  addlogos_text = NULL)

```

## Arguments

<code>name</code>	A English name, or alphanumeric, containing English alphabets, numbers, dots, dashes, arroww, colons, semicolons, comma among punctuations.
<code>colfill</code>	The color used for the symbol
<code>plot</code>	binary, if FALSE, returns only the co-ordinates of the symbol in the [0,1] X [0,1] grid, along with block id labels and their corresponding colors. If TRUE, plots the symbol with specified color in a new grid window.
<code>total_chars</code>	The total number of character symbols in the user library. The default is the default library provided by Logolas, but the user can add symbols that he creates to this list.
<code>addlogos</code>	Vector of additional logos/symbols defined by user
<code>addlogos_text</code>	Vector of the names given to the additional logos/symbols defined by user.

**Value**

Along with symbol plot, if plot is TRUE, returns a list with the following items.

x	X co-ordinates of the logo in the [0,1] X [0,1] grid window
y	Y co-ordinates of the logo in the [0,1] X [0,1] grid window
id	id vector representing blocks in the logo co-ordinates
fill	a vector equal to the number of distinct ids or blocks in the logo, whose elements correspond to colors of these blocks

**Examples**

```
makemylogo("KUSHAL")
cols = RColorBrewer::brewer.pal.info[RColorBrewer::brewer.pal.info$category == 'qual',]
col_vector = unlist(mapply(RColorBrewer::brewer.pal, cols$maxcolors, rownames(cols)))
makemylogo("Evening", plot=TRUE, colfill=col_vector)
```

# Index

## \*Topic **datasets**

himalayan\_fauna\_2\_clusters, [2](#)

himalayan\_fauna\_3\_clusters, [2](#)

himalayan\_fauna\_2\_clusters, [2](#)

himalayan\_fauna\_3\_clusters, [2](#)

ic\_computer, [3](#)

logomaker, [4](#)

makemylogo, [6](#)