

Package: desiR (via r-universe)

September 11, 2024

Type Package

Title Desirability Functions for Ranking, Selecting, and Integrating Data

Version 1.2.2

Date 2021-04-16

Author Stanley E. Lazic

Maintainer Stanley E. Lazic <stan.lazic@cantab.net>

Description Functions for (1) ranking, selecting, and prioritising genes, proteins, and metabolites from high dimensional biology experiments, (2) multivariate hit calling in high content screens, and (3) combining data from diverse sources.

License GPL-3

LazyData true

Depends R (>= 2.10)

Suggests knitr, rmarkdown

URL <https://github.com/stanlazic/desiR>

BugReports <https://github.com/stanlazic/desiR/issues>

VignetteBuilder knitr

RoxygenNote 7.1.1

Encoding UTF-8

Repository <https://stanlazic.r-universe.dev>

RemoteUrl <https://github.com/stanlazic/desir>

RemoteRef HEAD

RemoteSha 3c591b02668bc22909fdcf726188ef95aa7af927

Contents

d.4pl	2
d.central	3
d.ends	4
d.high	5
d.low	6
d.overall	7
d.rank	8
des.line	9
farmer2005	10

Index	12
--------------	-----------

d.4pl	<i>Four parameter logistic desirability function</i>
-------	--

Description

Maps a numeric variable to a 0-1 scale with a logistic function.

Usage

```
d.4pl(x, hill, inflec, des.min = 0, des.max = 1)
```

Arguments

x	Vector of numeric or integer values.
hill	Hill coefficient. It controls the steepness and direction of the slope. A value greater than zero has a positive slope and a value less than zero has a negative slope. The higher the absolute value, the steeper the slope.
inflec	Inflection point. Is the point on the x-axis where the curvature of the function changes from concave upwards to concave downwards (or vice versa).
des.min, des.max	The lower and upper asymptotes of the function. Defaults to zero and one, respectively.

Details

This function uses a four parameter logistic model to map a numeric variable onto a 0-1 scale. Whether high or low values are deemed desirable can be controlled with the `hill` parameter; when `hill > 0` high values are desirable and when `hill < 0` low values are desirable

Note that if the data contain both positive and negative values this function does not provide a monotonic mapping (see example).

Value

Numeric vector of desirability values.

See Also[d.low, d.high](#)**Examples**

```
# High values are desirable
x1 <- seq(80, 120, 0.01)
d1 <- d.4pl(x = x1, hill = 20, inflec = 100)
plot(d1 ~ x1, type="l")

# Low values are desirable (negative slope), with a minimum
# desirability of 0.3
d2 <- d.4pl(x = x1, hill = -30, inflec = 100, des.min=0.3)
plot(d2 ~ x1, type="l", ylim=c(0,1))

# Beware of how the function behaves when the data contain both
# positive and negative values
x2 <- seq(-20, 20, 0.01)
d3 <- d.4pl(x = x2, hill = 20, inflec = 1)
plot(d3 ~ x2, type="l")
```

d.central*Central values are desirable*

Description

Maps a numeric variable to a 0-1 scale such that values in the middle of the distribution are desirable.

Usage

```
d.central(x, cut1, cut2, cut3, cut4, des.min = 0, des.max = 1, scale = 1)
```

Arguments

x	Vector of numeric or integer values.
cut1, cut2, cut3, cut4	Values of the original data that define where the desirability function changes.
des.min, des.max	Minimum and maximum desirability values, defaults to zero and one, respectively.
scale	Controls how steeply the function increases or decreases.

Details

Values less than cut1 and greater than cut4 will have a low desirability. Values between cut2 and cut3 will have a high desirability. Values between cut1 and cut2 and between cut3 and cut4 will have intermediate values. This function is useful when extreme values are undesirable. For example, outliers or values outside of allowable ranges. If cut2 and cut3 are close to each other, this function can be used when a target value is desirable.

Value

Numeric vector of desirability values.

See Also

[d.ends](#)

Examples

```
set.seed(1)
x <- rnorm(1000, mean=100, sd =5) # generate data
d <- d.central(x, cut1=90, cut2=95, cut3=105, cut4=110, scale=1)

# plot data
hist(x, breaks=30)
# add line
des.line(x, "d.central", des.args=c(cut1=90, cut2=95, cut3=105,
cut4=110, scale=1))

hist(x, breaks=30)
des.line(x, "d.central", des.args=c(cut1=90, cut2=95, cut3=105,
cut4=110, des.min=0.1, des.max=0.95, scale=1.5))

# target value
hist(x, breaks=30)
des.line(x, "d.central", des.args=c(cut1=90, cut2=99.9, cut3=100.1, cut4=110))
```

d.ends

Extreme (both high and low) values are desirable

Description

Maps a numeric variable to a 0-1 scale such that values at the ends of the distribution are desirable.

Usage

```
d.ends(x, cut1, cut2, cut3, cut4, des.min = 0, des.max = 1, scale = 1)
```

Arguments

x	Vector of numeric or integer values.
cut1, cut2, cut3, cut4	Values of the original data that define where the desirability function changes.
des.min, des.max	Minimum and maximum desirability values. Defaults to zero and one, respectively.
scale	Controls how steeply the function increases or decreases.

Details

Values less than cut1 and greater than cut4 will have a high desirability. Values between cut2 and cut3 will have a low desirability. Values between cut1 and cut2 and between cut3 and cut4 will have intermediate values. This function is useful when the data represent differences between groups; for example, log2 fold-changes in gene expression. In this case, both high and low values are of interest.

Value

Numeric vector of desirability values.

See Also

[d.central](#)

Examples

```
set.seed(1)
x <- rnorm(1000, mean=100, sd =5) # generate data
d <- d.ends(x, cut1=90, cut2=95, cut3=105, cut4=110, scale=1)

# plot data
hist(x, breaks=30)
# add line
des.line(x, "d.ends", des.args=c(cut1=90, cut2=95, cut3=105,
cut4=110, scale=1))

hist(x, breaks=30)
des.line(x, "d.ends", des.args=c(cut1=90, cut2=95, cut3=105,
cut4=110, des.min=0.1, des.max=0.95, scale=1.5))
```

d.high

High values are desirable

Description

Maps a numeric variable to a 0-1 scale such that high values are desirable.

Usage

```
d.high(x, cut1, cut2, des.min = 0, des.max = 1, scale = 1)
```

Arguments

x Vector of numeric or integer values.
cut1, cut2 Values of the original data that define where the desirability function changes.

des.min, des.max Minimum and maximum desirability values. Defaults to zero and one, respectively.

scale Controls how steeply the function increases or decreases.

Details

Values less than cut1 will have a low desirability. Values greater than cut2 will have a high desirability. Values between cut1 and cut2 will have intermediate values.

Value

Numeric vector of desirability values.

See Also

[d.low](#), [d.4pl](#)

Examples

```
set.seed(1)
x <- rnorm(1000, mean=100, sd =5) # generate data
d <- d.high(x, cut1=90, cut2=110, scale=1)

# plot data
hist(x, breaks=30)
# add line
des.line(x, "d.high", des.args=c(cut1=90, cut2=110, scale=1))

hist(x, breaks=30)
des.line(x, "d.high", des.args=c(cut1=90, cut2=110, des.min=0.1,
des.max=0.95, scale=1.5))
```

d.low

Low values are desirable

Description

Maps a numeric variable to a 0-1 scale such that low values are desirable.

Usage

```
d.low(x, cut1, cut2, des.min = 0, des.max = 1, scale = 1)
```

Arguments

x	Vector of numeric or integer values.
cut1, cut2	Values of the original data that define where the desirability function changes.
des.min, des.max	Minimum and maximum desirability values. Defaults to zero and one, respectively.
scale	Controls how steeply the function increases or decreases.

Details

Values less than cut1 will have a high desirability. Values greater than cut2 will have a low desirability. Values between cut1 and cut2 will have intermediate values.

Value

Numeric vector of desirability values.

See Also

[d.high](#), [d.4pl](#)

Examples

```
set.seed(1)
x <- rnorm(1000, mean=100, sd =5) # generate data
d <- d.low(x, cut1=90, cut2=110, scale=1)

# plot data
hist(x, breaks=30)
# add line
des.line(x, "d.low", des.args=c(cut1=90, cut2=110, scale=1))

hist(x, breaks=30)
des.line(x, "d.low", des.args=c(cut1=90, cut2=110, des.min=0.1,
des.max=0.95, scale=1.5))
```

d.overall

Combine individual desirabilities

Description

Combines any number of desirability values into an overall desirability.

Usage

```
d.overall(..., weights = NULL)
```

Arguments

... Any number of individual desirabilities.

weights Allows some desirabilities to count for more in the overall calculation. Defaults to equal weighting.

Details

This function takes any number of individual desirabilities and combines them with a weighted geometric mean to give an overall desirability. The weights should be chosen to reflect the importance of the variables. The values of the weights do not matter, only their relative differences. Therefore weights of 4, 2, 1 are the same as 1, 0.5, 0.25. In both cases the second weight is half of the first, and the third weight is a quarter of the first.

Value

Numeric vector of desirability values.

Examples

```
set.seed(1)
x1 <- rnorm(1000, mean=100, sd =5) # generate data
x2 <- rnorm(1000, mean=100, sd =5)

d1 <- d.high(x1, cut1=90, cut2=110, scale=1)
d2 <- d.low(x2, cut1=90, cut2=110, scale=1)

D <- d.overall(d1, d2, weights=c(1, 0.5))
plot(rev(sort(D)), type="l")
```

d.rank

Converts values to ranks, then ranks to desirabilities

Description

Values are ranked from low to high or high to low, and then the ranks are mapped to a 0-1 scale.

Usage

```
d.rank(x, low.to.high, ties = "min")
```

Arguments

x Vector of numeric or integer values.

low.to.high If TRUE, low ranks have high desirabilities; if FALSE, high ranks have high desirabilities.

ties Specifies how to deal with ties in the data. The value is passed to the 'ties.method' argument of the rank() function. Default is 'min'. See help(rank) for more information.

Details

If low values of a variable are desirable (e.g. p-values) set the argument `low.to.high=TRUE`, otherwise `low.to.high=FALSE`.

If extreme values in either direction are of interest (e.g. fold-changes), take the absolute value of the variable and use `low.to.high=FALSE`. See the example below.

This function is less flexible than the others but it can be used to compare the desirability approach with rank aggregation methods.

Value

Numeric vector of desirability values.

Examples

```
set.seed(1)
x1 <- rnorm(1000, mean=100, sd =5) # generate data
d <- d.rank(x1, low.to.high=TRUE)

# plot data
hist(x1, breaks=30)
# add line
des.line(x1, "d.rank", des.args=c(low.to.high=TRUE))

x2 <- rnorm(1000, mean=0, sd =5) # positive and negative values
# could be fold-changes, mean differences, or t-statistics
hist(abs(x2), breaks=30)
# add line
des.line(abs(x2), "d.rank", des.args=c(low.to.high=FALSE))
```

des.line

Plots a desirability function on an existing graph

Description

Plots any of the desirability functions on top of a graph, usually a histogram or density plot.

Usage

```
des.line(x, des.func, des.args, ...)
```

Arguments

<code>x</code>	Vector of numeric or integer values.
<code>des.func</code>	Name of the desirability function to plot (in quotes).
<code>des.args</code>	A vector of named arguments for the chosen desirability function.
<code>...</code>	Arguments for the plotting function (e.g. <code>xlim</code> , <code>lwd</code> , <code>lty</code>).

Details

This function can be used to visualise how the desirabilities are mapped from the raw data to a 0-1 scale, which can help select suitable cut points. The scale of the y-axis has a minimum of 0 and a maximum of 1.

WARNING: If you set `xlim` values for the histogram or density plot, then you must pass the same `xlim` values to `des.line`; otherwise the data and desirability function (plotted line) will be misaligned. If `xlim` is not set, then the same default values will be used for the data and the function.

Value

Plotted values of the desirability function.

See Also

[d.low](#), [d.high](#), [d.central](#), [d.ends](#), [d.4pl](#)

Examples

```
set.seed(1)
x1 <- rnorm(100, 10, 2)
hist(x1, breaks=10)
des.line(x1, "d.high", des.args=c(cut1=10, cut2=11))
des.line(x1, "d.high", des.args=c(cut1=10, cut2=11,
des.min=0.1, scale=0.5))
```

farmer2005

Breast cancer microarray dataset

Description

1000 randomly selected probesets from a breast cancer microarray dataset (Farmer et al., 2005).

Format

A data frame with 1000 rows and 7 variables:

ProbeSet: Affymetrix probesets from the U133A chip.

GeneID: Gene symbol.

logFC: Log2 fold change for the basal versus luminal comparison.

AveExpr: Mean expression across all samples.

P.Value: P-value for basal versus luminal comparison.

SD: Standard deviation across all samples.

PCNA.cor: Correlation with PCNA (a marker of proliferating cells).

Details

These data are the results from an analysis comparing the basal and luminal samples. The apocrine samples are excluded.

References

Farmer P, Bonnefoi H, Becette V, Tubiana-Hulin M, Fumoleau P, Larsimont D, Macgrogan G, Bergh J, Cameron D, Goldstein D, Duss S, Nicoulaz AL, Brisken C, Fiche M, Delorenzi M, Iggo R. Identification of molecular apocrine breast tumours by microarray analysis. *Oncogene*. 2005 24(29):4660-4671.

Index

d.4pl, [2](#), [6](#), [7](#), [10](#)
d.central, [3](#), [5](#), [10](#)
d.ends, [4](#), [4](#), [10](#)
d.high, [3](#), [5](#), [7](#), [10](#)
d.low, [3](#), [6](#), [6](#), [10](#)
d.overall, [7](#)
d.rank, [8](#)
des.line, [9](#)

farmer2005, [10](#)