

Package: sinaplot (via r-universe)

September 12, 2024

Type Package

Title An Enhanced Chart for Simple and Truthful Representation of Single Observations over Multiple Classes

Version 1.1.1

Date 2017-09-06

Maintainer Nikos Sidiropoulos <nikos.sidiro@gmail.com>

Description The sinaplot is a data visualization chart suitable for plotting any single variable in a multiclass data set. It is an enhanced jitter strip chart, where the width of the jitter is controlled by the density distribution of the data within each class.

Depends R(>= 3.1.0), plyr(>= 1.8.4)

Suggests rmarkdown, knitr, RColorBrewer

License GPL (>=2)

LazyData TRUE

VignetteBuilder knitr

RoxygenNote 6.0.1

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

RemoteUrl <https://github.com/sidiropoulos/sinaplot>

RemoteRef HEAD

RemoteSha 6aa060dae1889bad29cf9d37817e8e2a31284c2f

Contents

blood	2
sinaplot	2

Index	6
--------------	----------

blood	<i>Expression data from 2095 AML/ALL and healthy bone marrow cells.</i>
-------	-------------------------------------------------------------------------

Description

Expression data from 2095 AML/ALL and healthy bone marrow cells.

Usage

```
data(blood)
```

Format

A data frame with 2095 rows and 2 columns (Class (AML/ALL subtype), Gene expression values).

Source

```

http://servers.binf.ku.dk/bloodspot/
http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE13159
http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE15434
http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE61804
http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE14468
http://cancergenome.nih.gov/

```

sinaplot	<i>sinaplot</i>
----------	-----------------

Description

The SinaPlot is a data visualization chart suitable for plotting any single variable in a multiclass dataset. It is an enhanced jitter strip chart, where the width of the jitter is controlled by the density distribution of the data within each class.

Usage

```

sinaplot(x, ...)

## Default S3 method:
sinaplot(x, groups = NULL, method = c("density",
  "counts"), scale = TRUE, adjust = 0.75, bins = 50, bin_limit = 1,
  maxwidth = 1, seed = NULL, plot = TRUE, add = FALSE, log = FALSE,
  labels = NULL, xlab = "", ylab = "", col = NULL, pch = NULL, ...)

## S3 method for class 'formula'
sinaplot(formula, data = NULL, ..., subset,
  na.action = NULL, xlab, ylab)

```

Arguments

x	numeric vector or a data frame or a list of numeric vectors to be plotted.
...	arguments to be passed to <code>plot</code> .
groups	optional vector of length(x).
method	choose the method to spread the samples within the same bin along the x-axis. Available methods: "density" and "counts". See Details.
scale	a logical that indicates whether the width of each group should be scaled relative to the group with the highest density. Default: TRUE.
adjust	adjusts the bandwidth of the density kernel when <code>method == "density"</code> (see density).
bins	number of bins to divide the y-axis into when <code>method == "counts"</code> . Default: 50.
bin_limit	if the samples within the same y-axis bin are more than <code>bin_limit</code> , the samples's X coordinates will be adjusted.
maxwidth	control the maximum width the points can spread into. Values between 0 and 1.
seed	a single value that controls the random sample jittering. Set to an integer to enable plot reproducibility. Default NULL.
plot	logical. When TRUE the sinaplot is produced, otherwise the function returns the new sample coordinates. Default: TRUE.
add	logical. If true add boxplot to current plot.
log	logical. If true it uses a logarithmic scale on the y-axis.
labels	labels for each group. Recycled if necessary. By default, these are inferred from the data.
xlab, ylab	axis labels.
pch, col	plotting characters and colors, specified by group. Recycled if necessary.
formula	a formula, such as <code>y ~ grp</code> , where y is a numeric vector of data values to be split into groups according to the grouping variable <code>grp</code> (usually a factor).
data	a data.frame (or list) from which the variables in formula should be taken.
subset	an optional vector specifying a subset of observations to be used for plotting.
na.action	a function which indicates what should happen when the data contain NAs. The default is to ignore missing values in either the response or the group.

Details

There are two available ways to define the x-axis borders for the samples to spread within:

- `method = "density"`
A density kernel is estimated along the y-axis for every sample group. The borders are then defined by the density curve. Tuning parameter `adjust` can be used to control the density bandwidth in the same way it is used in [density](#).
- `method = "counts"`:
The borders are defined by the number of samples that occupy the same bin and the parameter `maxwidth` in the following fashion:
`xBorder = nsamples * maxwidth`

Value

x	discrete x-coordinates, split by group
y	input values
group	input groups
scaled	final x-coordinates, adjusted by sinaplot
NULL	
NULL	

Examples

```
## sinaplot on a formula:

data("blood", package = "sinaplot")
boxplot(Gene ~ Class, data = blood)
sinaplot(Gene ~ Class, data = blood, pch = 20, add = TRUE)

## sinaplot on a data.frame:

df <- data.frame(Uni05 = (1:100)/21, Norm = rnorm(100),
                 `5T` = rt(100, df = 5), Gam2 = rgamma(100, shape = 2))
boxplot(df)
sinaplot(df, add = TRUE, pch = 20)

## sinaplot on a list:

bimodal <- c(rnorm(300, -2, 0.6), rnorm(300, 2, 0.6))
uniform <- runif(500, -4, 4)
normal <- rnorm(800,0,3)

distributions <- list(uniform = uniform, bimodal = bimodal, normal = normal)
boxplot(distributions, col = 2:4)
sinaplot(distributions, add = TRUE, pch = 20)

## sinaplot on a vector:

x <- c(rnorm(200, 4, 1), rnorm(200, 5, 2), rnorm(400, 6, 1.5))
groups <- c(rep("Cond1", 200), rep("Cond2", 200), rep("Cond3", 400))

sinaplot(x, groups)

par(mfrow = c(2, 2))

sinaplot(x, groups, pch = 20, col = 2:4)
sinaplot(x, groups, scale = FALSE, pch = 20, col = 2:4)
sinaplot(x, groups, scale = FALSE, adjust = 1/6, pch = 20, col = 2:4)
sinaplot(x, groups, scale = FALSE, adjust = 3, pch = 20, col = 2:4)

#blood

par(mfrow = c(1,1))
```

```
sinaplot(blood$Gene, blood$Class)

old.mar <- par()$mar
par(mar = c(9,4,4,2) + 0.1)
groups <- levels(blood$Class)

sinaplot(blood$Gene, blood$Class, pch = 20, xaxt = "n", col = rainbow(18))
axis(1, at = 1:length(groups), labels = FALSE)
text(1:length(groups), y = par()$usr[3] - 0.1 * (par()$usr[4] - par()$usr[3]),
     xpd = TRUE, srt = 45, adj = 1, labels = groups)
par(mar = old.mar)
```

Index

- * **datasets**
 - blood, [2](#)
- blood, [2](#)
- density, [3](#)
- plot, [3](#)
- sinaplot, [2](#)