

# Package ‘areaplot’

November 20, 2023

**Version** 2.1.1

**Date** 2023-11-20

**Title** Plot Stacked Areas and Confidence Bands as Filled Polygons

**Imports** graphics, grDevices, stats

**Suggests** MASS

**Description** Plot stacked areas and confidence bands as filled polygons, or add polygons to existing plots. A variety of input formats are supported, including vectors, matrices, data frames, formulas, etc.

**License** GPL-3

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**NeedsCompilation** no

**Author** Arni Magnusson [aut, cre]

**Maintainer** Arni Magnusson <thisisarni@gmail.com>

**Repository** CRAN

**Date/Publication** 2023-11-20 08:20:02 UTC

## R topics documented:

areaplot-package . . . . .	2
areaplot . . . . .	2
confplot . . . . .	4
<b>Index</b>	<b>7</b>

---

areaplot-package      *Plot Stacked Areas and Confidence Bands as Filled Polygons*

---

### Description

Plot stacked areas and confidence bands as filled polygons, or add polygons to existing plots. A variety of input formats are supported, including vectors, matrices, data frames, formulas, etc.

### Details

*Plot:*

<code>areaplot</code>	stacked area
<code>confplot</code>	confidence bands

### Author(s)

Arni Magnusson.

---

areaplot      *Area Plot*

---

### Description

Produce a stacked area plot, or add polygons to an existing plot.

### Usage

```
areaplot(x, ...)
```

## Default S3 method:

```
areaplot(x, y = NULL, prop = FALSE, rev = FALSE,
  add = FALSE, xlab = NULL, ylab = NULL, col = NULL, legend = FALSE,
  args.legend = NULL, ...)
```

## S3 method for class 'formula'

```
areaplot(formula, data, subset, na.action, xlab = NULL,
  ylab = NULL, ...)
```

### Arguments

<code>x</code>	a numeric vector of x values, or if <code>y=NULL</code> a numeric vector of y values. Can also be a 1-dimensional table (x values in names, y values in array), matrix or 2-dimensional table (x values in row names and y values in columns), a data frame (x values in first column and y values in subsequent columns), or a time-series object of class <code>ts/mts</code> .
----------------	---

...	further arguments passed to <code>areaplot.default</code> , <code>matplot</code> , and <code>polygon</code> .
<code>y</code>	a numeric vector of y values, or a matrix containing y values in columns.
<code>prop</code>	whether data should be plotted as proportions, so stacked areas equal 1.
<code>rev</code>	whether to plot the stacked areas from bottom to top, instead of top to bottom.
<code>add</code>	whether polygons should be added to an existing plot.
<code>xlab</code>	a label for x axis.
<code>ylab</code>	a label for y axis.
<code>col</code>	fill color of polygon(s). The default is a vector of gray colors.
<code>legend</code>	a logical indicating whether a legend should be added, or a vector of strings for the legend. This only applies when more than one series is plotted.
<code>args.legend</code>	a list of additional arguments to pass to the <code>legend</code> function.
<code>formula</code>	a <a href="#">formula</a> , such as <code>y~x</code> , <code>cbind(y1,y2)~x</code> , or <code>y~x+group</code> , specifying x and y values. A dot on the left-hand side, <code>.~x</code> , means all variables except the one specified on the right-hand side.
<code>data</code>	a data frame (or list) from which the variables in formula should be taken.
<code>subset</code>	an optional vector specifying a subset of observations to be used.
<code>na.action</code>	a function which indicates what should happen when the data contain NA values. The default is to ignore missing values in the given variables.

**Value**

Matrix of cumulative sums that was used for plotting.

**See Also**

[polygon](#) is the underlying function used to draw polygons.

[confplot](#) plots confidence bands as a filled area.

[areaplot-package](#) gives an overview of the package.

**Examples**

```
areaplot(rpois(10,40))
areaplot(rnorm(10))

# formula
areaplot(Armed.Forces~Year, data=longley)
areaplot(cbind(Armed.Forces,Unemployed)~Year, data=longley)
areaplot(.~Year, data=longley)
areaplot(circumference~age+Tree, Orange)

# add=TRUE
plot(1940:1970, 500*runif(31), ylim=c(0,500))
areaplot(Armed.Forces~Year, data=longley, add=TRUE)

# data frame
mydata <- longley[c("Year", "GNP")]
```

```

areaplot(mydata)

# matrix
areaplot(WorldPhones)
areaplot(WorldPhones, prop=TRUE)

# table
require(MASS)
areaplot(table(Aids2$age))
areaplot(table(Aids2$age, Aids2$sex))

# ts/mts
areaplot(austres)
areaplot(Seatbelts[,c("drivers", "front", "rear")],
        ylab="Killed or seriously injured")
abline(v=1983+1/12, lty=3)

# legend
require(MASS)
areaplot(table(Aids2$age, Aids2$sex), legend=TRUE, col=c(2,4))
areaplot(table(Aids2$age, Aids2$sex), legend=TRUE, col=c(2,4), rev=TRUE)
wp <- WorldPhones[,order(colnames(WorldPhones))]
areaplot(wp, col=2:8, legend=TRUE, args.legend=list(x="topleft"))
areaplot(wp, col=2:8, legend=TRUE, args.legend=list(x="topleft"), rev=TRUE)

```

---

confplot

*Plot Confidence Bands*

---

## Description

Plot confidence bands of lower and upper y values as a filled area, or add polygon to an existing plot.

## Usage

```

confplot(x, ...)

## Default S3 method:
confplot(x, y1 = NULL, y2 = NULL, add = FALSE,
        xlab = NULL, ylab = NULL, border = NA, col = "lightgray", ...)

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

```

## Arguments

**x** a numeric vector of x values. Alternatively, x can be a matrix or data frame containing x values in the first column and lower and upper y values in the next two columns.

...	further arguments passed to <code>confplot.default</code> , <code>matplot</code> , and <code>polygon</code> .
<code>y1</code>	a numeric vector of lower y values. Alternatively, <code>y1</code> can be a matrix or data frame containing lower and upper y values in two columns.
<code>y2</code>	a numeric vector of upper y values, if not already supplied in <code>x</code> or <code>y1</code> .
<code>add</code>	whether confidence bands should be added to an existing plot.
<code>xlab</code>	a label for x axis.
<code>ylab</code>	a label for y axis.
<code>border</code>	border color of polygon. The default NA is to omit borders.
<code>col</code>	fill color of polygon.
<code>formula</code>	a <a href="#">formula</a> , such as <code>cbind(y1,y2)~x</code> , specifying x and y values.
<code>data</code>	a data frame (or list) from which the variables in formula should be taken.
<code>subset</code>	an optional vector specifying a subset of observations to be used.
<code>na.action</code>	a function which indicates what should happen when the data contain NA values. The default is to ignore missing values in the given variables.

**Value**

Data frame of coordinates that were used for plotting.

**See Also**

[polygon](#) is the underlying function used to draw polygons.

[areaplot](#) produces a stacked area plot.

[areaplot-package](#) gives an overview of the package.

The [gplots](#) and [plotrix](#) packages provide functions to plot error bars.

**Examples**

```

model <- lm(log(dist)~log(speed), cars)
ci95 <- predict(model, data.frame(speed=4:25), interval="confidence")
ci50 <- predict(model, data.frame(speed=4:25), interval="confidence", level=0.5)
x <- log(4:25)
y1 <- ci95[,"lwr"]
y2 <- ci95[,"upr"]
mydata <- data.frame(x, y1, y2)

# Input format
confplot(x, y1, y2)           # vectors
confplot(x, cbind(y1,y2))    # y values in 2 columns
confplot(mydata)             # data in 3 columns
confplot(cbind(y1,y2)~x, mydata) # formula

# Overlay
plot(log(dist)~log(speed), cars, type="n")
confplot(x, ci95[,2:3], add=TRUE)
confplot(x, ci50[,2:3], add=TRUE, col="darkgray")

```

```
lines(x, ci95[,1])  
points(log(dist)~log(speed), cars)
```

# Index

areaplot, [2](#), [2](#), [5](#)  
areaplot-package, [2](#)  
confplot, [2](#), [3](#), [4](#)  
formula, [3](#), [5](#)  
legend, [3](#)  
polygon, [3](#), [5](#)