Package 'fastGraph'

July 7, 2016

Type Package

Title Fast Drawing and Shading of Graphs of Statistical Distributions

Version 1.1

Date 2016-07-07

Author Steven T. Garren [aut, cre]

Maintainer Steven T. Garren <garrenst@jmu.edu>

Description Provides functionality to produce graphs of probability density functions and cumulative distribution functions with few keystrokes, allows shading under the curve of the probability density function to illustrate concepts such as p-values and critical values, and fits a simple linear regression line on a scatter plot with the equation as the main title.

License GPL-3

Depends R (>= 2.0)

NeedsCompilation no

Repository CRAN

Date/Publication 2016-07-07 20:24:49

R topics documented:

`fastGraph-package` *Fast Drawing and Shading of Graphs of Statistical Distributions*

Description

Provides functionality to produce graphs of probability density functions and cumulative distribution functions with few keystrokes, allows shading under the curve of the probability density function to illustrate concepts such as p-values and critical values, and fits a simple linear regression line on a scatter plot with the equation as the main title.

Details

Package:	fastGraph
Type:	Package
Version:	1.1
Date:	2016-07-07
License:	GPL-3

- `plotDist` draws as many as three probability density functions or cumulative distribution functions on the same graph.
- `shadeDist` draws a probability density function, shades in area under the curve, and lists the probability in the title of the graph.
- `shadePhat` is similar to `shadeDist` but considers the distribution of only the sample proportion.
- `plotLine` performs a simple scatter plot, fits the linear regression line, and states the equation of the line in the title.
- `getMinMax` is called by both `plotDist` and `shadeDist` for determining a reasonable domain for plotting the graph.

Author(s)

Steven T. Garren, James Madison University, Harrisonburg, Virginia, USA; GARRENST AT JMU DOT EDU

See Also

`plot` and `lm`

Examples

```
# Plots cumulative distribution functions of a standard normal, a central t with 4 d.f.,
# and a t with 4 d.f. and non-centrality parameter = 1.3 in black, red, and green, respectively.
plotDist( "pnorm", 0, 1, "pt", 4, 0, "pt", 4, 1.3 )
```

```
# Plots Binomial(n=100,p) density functions, where p=0.4, 0.5, and 0.7
# in colors black, red, and green, respectively.
plotDist( "dbinom", 100, 0.4, "dbinom", 100, 0.5, "dbinom", 100, 0.7 )

# Shows P(|T| > 1.8), where T is t distributed with 9 d.f.
shadeDist( c(-1.8, 1.8), "dt", 9 )

# Shows P(X > 8), where X ~ Poisson(mean=6).
shadeDist( 8, "dpois", 6, lower.tail=FALSE, col=c("purple","green") )

# Graphs line of simple linear regression model and states equation.
plotLine( c(-5,6,2,9,-11), c(-7,17,21,29,8), digits.intercept=3, digits.slope=4 )

# Finds a reasonable domain when plotting a Normal(mean=20, sd=5) density function.
getMinMax( , , "dnorm", 20, 5 )
```

`getMinMax` *Finds a Reasonable Domain for Plotting a Graph*

Description

This function computes a reasonable domain for plotting one, two, or three distribution functions by truncating small tail probabilities. This function also lists the population medians.

Usage

```
getMinMax(xmin = NULL, xmax = NULL, distA, parmA1 = NULL, parmA2 = NULL, distB = NULL,
        parmB1 = NULL, parmB2 = NULL, distC = NULL, parmC1 = NULL, parmC2 = NULL)
```

Arguments

`xmin`	A lower bound, usually set to `NULL` during input.
`xmax`	An upper bound, usually set to `NULL` during input.
`distA`	Character variable naming the first probability density function (starting with "d") or cumulative density function (starting with "p").
`parmA1`	The first argument in `distA`, excluding the dummy argument. Alternatively, `parmA1` may be set to be a vector of arguments, excluding the dummy argument.
`parmA2`	The second argument in `distA`, excluding the dummy argument. Alternatively, `parmA2` may be set to be a vector of arguments, excluding the dummy argument and `parmA1`.
`distB`	Character variable naming the second probability density function (starting with "d") or cumulative density function (starting with "p").
`parmB1`	The first argument in `distB`, excluding the dummy argument. Alternatively, `parmB1` may be set to be a vector of arguments, excluding the dummy argument.
`parmB2`	The second argument in `distB`, excluding the dummy argument. Alternatively, `parmB2` may be set to be a vector of arguments, excluding the dummy argument and `parmB1`.

distC	Character variable naming the third probability density function (starting with "d") or cumulative density function (starting with "p").
parmC1	The first argument in distC, excluding the dummy argument. Alternatively, parmC1 may be set to be a vector of arguments, excluding the dummy argument.
parmC2	The second argument in distC, excluding the dummy argument. Alternatively, parmC2 may be set to be a vector of arguments, excluding the dummy argument and parmC1.

Details

This function getMinMax is automatically called by plotDist and shadeDist, so the user does not actually need to directly call getMinMax when executing plotDist and shadeDist. This function by itself does not construct a graph.

Value

xmin	A reasonable value of a lower bound for the domain of a graph.
xmax	A reasonable value of an upper bound for the domain of a graph.
medianA	The population median of distA.
medianB	The population median of distB.
medianC	The population median of distC.

Author(s)

Steven T. Garren, James Madison University, Harrisonburg, Virginia, USA

See Also

plotDist and shadeDist

Examples

```
getMinMax(  , , "dnorm", 20, 5  ) # Normal(mu=20, sigma=5)

# Standard normal, and t with 4 degrees of freedom
getMinMax(  , , "dnorm", 0, 1, "dt", 4, 0 )

# Standard normal, central t with 4 d.f., and t with 4 d.f. and non-centrality parmater = 1.2
getMinMax(  , , "dnorm", 0, 1, "dt", 4, 0, "dt", 4, 1.2 )

# Force minimum to be -3.
getMinMax( -3, , "dnorm", 0, 1 )

# Force maximum to be 3.
getMinMax( , 3, "dnorm", 0, 1 )
```

plotDist *Plotting of Statistical Distributions*

Description

This function plots as many as three probability density functions and cumulative distribution functions on the same graph using just one command, where the domain of the graph need not be specified by the user.

Usage

```
plotDist(distA = "dnorm", parmA1 = NULL, parmA2 = NULL, distB = NULL, parmB1 = NULL,
         parmB2 = NULL, distC = NULL, parmC1 = NULL, parmC2 = NULL, xlab = NULL,
         xmin = NULL, xmax = NULL, col = c("black", "red", "darkgreen"),
         is.discrete = NULL, additional.x.range = NULL, lwd = 5, ...)
```

Arguments

distA	Character variable naming the first probability density function (starting with "d") or cumulative density function (starting with "p") to be graphed.
parmA1	The first argument in distA, excluding the dummy argument. For example, if distA="dnorm", then parmA1 is the mean from "dnorm". Alternatively, parmA1 may be set to be a vector of arguments, excluding the dummy argument.
parmA2	The second argument in distA, excluding the dummy argument. For example, if distA="dnorm", then parmA2 is the sd from "dnorm". Alternatively, parmA2 may be set to be a vector of arguments, excluding both the dummy argument and parmA1.
distB	Character variable naming the second probability density function (starting with "d") or cumulative density function (starting with "p") to be graphed.
parmB1	The first argument in distB, excluding the dummy argument. Alternatively, parmB1 may be set to be a vector of arguments, excluding the dummy argument.
parmB2	The second argument in distB, excluding the dummy argument. Alternatively, parmB1 may be set to be a vector of arguments, excluding both the dummy argument and parmB1.
distC	Character variable naming the third probability density function (starting with "d") or cumulative density function (starting with "p") to be graphed.
parmC1	The first argument in distC, excluding the dummy argument. Alternatively, parmC1 may be set to be a vector of arguments, excluding the dummy argument.
parmC2	The second argument in distC, excluding the dummy argument. Alternatively, parmC2 may be set to be a vector of arguments, excluding both the dummy argument and parmC1.
xlab	The label of the x variable.
xmin	The minimum x-value to be graphed.
xmax	The maximum x-value to be graphed.

col	A vector specifying the colors of distA, distB, and distC, respectively.
is.discrete	A vector with 1, 2, or 3 logical values, indicating whether or not distA, distB, and distC are discrete. For built-in density functions, such as dbinom, pbinom, dgeom, pgeom, dhyper, phyper, dpois, ppois, dnbinom, and pnbinom, this argument is.discrete can be set to NULL, which is the default.
additional.x.range	A vector of two additional x-values for evaluating the function. This argument would be needed only if the user is dissatified with the domain determined by the function.
lwd	The line width for discrete distributions.
...	Optional arguments to be passed to the plot function (see par).

Details

If only one graph is to be plotted, then use distA. If only two graphs are to be plotted, then use distA and distB.

The arguments in plotDist are typically entered as first distribution plus two parameters, second distribution plus two parameters, and third distribution plus two parameters. If only one parameter of the distribution is needed, then the second parameter can be left as the default of NULL. If three or more parameters of the distribution are needed, then the first parameter can be assigned to be a vector consisting of all of the parameters.

The defaul value of distA is "dnorm"; i.e., for plotting the normal distribution.

The default values of all of the arguments following parmC2 usually are sufficient.

Note

This function plotDist calls functions getMinMax, plot, and curve.

Author(s)

Steven T. Garren, James Madison University, Harrisonburg, Virginia, USA

See Also

shadeDist, plot, and getMinMax

Examples

```
# Plots standard normal density in black, t density with 3 d.f. in red, and
# non-central t density with 3 d.f. and non-centrality parameter=1.4 in green.
plotDist( "dnorm", 0, 1, "dt", 3, 0, "dt", 3, 1.4 )

# Plots normal approximation to Binomial(n=100, p=0.4).
plotDist( , 40, sqrt(100*0.4*0.6), "dbinom", 100, 0.4, col=c("blue", "hotpink") )

# Plots cumulative normal and Cauchy distribution functions.
plotDist( "pnorm", 50, 10, "pcauchy", 50, 10, col=c("purple","orange") )
```

plotLine *X-Y Plotting with Simple Linear Regression Line and Equation*

Description

The function plots a simple scatter plot, fits the regression line on the scatter plot, and lists the equation of the fitted regression line as the title.

Usage

```
plotLine(x, y = NULL, data = NULL, xlab = NULL, ylab = NULL, pch = 19,
       col = c("black", "red"), digits.intercept = NULL, digits.slope = NULL, ...)
```

Arguments

x	The x coordinates of points in the plot. Alternatively, a single plotting structure or function can be provided.
y	The y coordinates of points in the plot, optional if x is an appropriate structure.
data	A data frame including the x and y coordinates.
xlab	The label of the x variable.
ylab	The label of the y variable.
pch	The plotting character; i.e., symbol to use. This can be either a single character or an integer code for one of a set of graphics symbols.
col	A vector of size two for the color code or name. The first value is the color of the plotting character, and the second value is the color of the fitted regression line.
digits.intercept	The desired number of significant digits for the intercept.
digits.slope	The desired number of significant digits for the slope.
...	Optional arguments to be passed to the plot function (see par).

Note

This function plotLine uses functions plot and lm.

Author(s)

Steven T. Garren, James Madison University, Harrisonburg, Virginia, USA

See Also

plot and lm

Examples

```
x <- c( 2, 6, 5, -3, 11, 3 )
y <- c( 16, 12, 19, -13, 27, 5 )
plotLine( x, y )

d <- data.frame( x=c( 2, 7, 9, 15, 12 ), y=c( 45, 32, 22, 15, 19 ) )
plotLine( y~x, data=d, col=c("blue","orange") )

plotLine( y~x, data=d, xlab="TIME", ylab="EXPENSE", digits.intercept=3, digits.slope=4 )
```

---

shadeDist *Displays Area Under Curve of Probability Density Function*

Description

This function plots a probability density function, shades the area under the curve, and computes the probability.

Usage

```
shadeDist(xshade = NULL, ddist = "dnorm", parm1 = NULL, parm2 = NULL, lower.tail = TRUE,
          xlab=NULL, xmin = NULL, xmax = NULL, xtic = TRUE, digits.prob = 4,
        digits.xtic = 3, is.discrete = NULL, additional.x.range = NULL, main = NULL,
          col = c("black", "hotpink"), lwd = 5, ...)
```

Arguments

xshade	A single number or vector of two numbers, denoting values on the x-axis where shading under the curve begins and ends.
ddist	Character variable naming the probability density function to be graphed. May be set to "dprob" for a sample proportion, using the same arguments as "dbinom".
parm1	The first argument in ddist, excluding the dummy argument. For example, if ddist="dnorm", then parm1 is the mean from "dnorm". Alternatively, parm1 may be set to be a vector of arguments, excluding the dummy argument.
parm2	The second argument in ddist, excluding the dummy argument. For example, if ddist="dnorm", then parm2 is the sd from "dnorm". Alternatively, parm2 may be set to be a vector of arguments, excluding both the dummy argument and parm1.
lower.tail	Logical; if TRUE (default), the lowest region is shaded; otherwise, the next lowest region is shaded.
xlab	The label of the x variable.
xmin	The minimum x-value to be graphed.
xmax	The maximum x-value to be graphed.

xtic	Logical or a vector of numbers. If xtic is TRUE (default), then the numbers on the x-axis include the median and xshade. If xtic is TRUE, then the default numbers from plot are listed on the x-axis. If xtic is a vector of numbers, then these numbers are listed on the x-axis.
digits.prob	The number of significant digits listed in the probability.
digits.xtic	The number of significant digits listed on the x-axis.
is.discrete	Logical; indicating whether or not the distribution is discrete. If is.discrete is NULL, then shadeDist automatically makes the correct choice for density functions already named in the stats package.
additional.x.range	A vector of two additional x-values for evaluating the function. This argument would be needed only if the user is dissatified with the domain determined by the function. This argument is ignored if ddist="dprob".
main	The main title given for the graph.
col	A vector of size two, specifying the colors of the density curve and the shading, respectively.
lwd	The line width for discrete distributions.
...	Optional arguments to be passed to the plot function (see par).

Details

When illustrating a left-sided p-value or any other left-sided probability, xshade should be a single number and set lower.tail=TRUE (default). When illustrating a right-sided p-value or any other right-sided probability, xshade should be a single number and set lower.tail=FALSE. When illustrating a two-sided p-value or any other two-sided probability, xshade should be a vector of two numbers and set lower.tail=TRUE (default). When illustrating the complement of a two-sided p-value or the complement of any other two-sided probability, xshade should be a vector of two numbers and set lower.tail=FALSE.

Note

This function shadeDist calls functions getMinMax, plot, and curve.

Author(s)

Steven T. Garren, James Madison University, Harrisonburg, Virginia, USA

See Also

plotDist and shadePhat

Examples

```
shadeDist( 1.96 ) # Shows P(Z<1.96) where Z ~ N(0,1)
shadeDist( 1.96, "dnorm", 0, 1 ) # Same as above.

shadeDist( 1.96, lower.tail = FALSE ) # Shows P(Z>1.96) where Z ~ N(0,1)
```

```
shadeDist( c( 40, 60 ), , 50, 10, lower.tail = FALSE ) # Shows P(40<X<60) where X~N(mu=50,sigma=10)

# Tail probabilities.
shadeDist( c( 40, 60 ), "dnorm", 50, 10, col=c("purple", "lightgreen") )

shadeDist( 5.8, "dchisq", 4, lower.tail = FALSE ) # Chi-squared distribution with 4 d.f.

shadeDist( c( -1.3, 1.3 ), "dt", 13 ) # t with 13 d.f.

shadeDist( 1.73, "dt", 15, lower.tail = FALSE, col=c("blue", "red") ) # t with 15 d.f.

shadeDist( 1.73, "dt", 15, 3, lower.tail = FALSE ) # t with 15 d.f. and non-centrality parameter=3

# P(X>6) where X~Poisson(mu=3.2)
shadeDist( 6, "dpois", 3.2, lower.tail = FALSE, col=c("brown", "yellow") )

shadeDist( c( 2, 5 ), "dbinom", 10, 0.4 ) # Binomial(n=10, p=0.4) probabilities

# Probabilities for sample proportion, phat = X/n, where X~Binomial(n=10, p=0.4)
shadeDist( 0.3, "dprop", 20, 0.4 )
```

---

shadePhat *Displays Area Under Curve of Probability Density Function of Sample Proportion*

Description

This function plots a probability density function of a sample proportion, shades the area under the curve, and computes the probability.

Usage

```
shadePhat(xshade = NULL, size = 1, prob = 0.5, lower.tail = TRUE, xmin = 0, xmax = 1,
        xlab = expression(hat(p)), xtic = TRUE, digits.prob = 4, digits.xtic = 3,
          main = NULL, col = c("black", "hotpink"), lwd = 5, ...)
```

Arguments

xshade	A single number or vector of two numbers, denoting values on the x-axis where shading under the curve begins and ends.
size	Number of Bernoulli trials (one or more).
prob	Probability of Bernoulli success.
lower.tail	Logical; if TRUE (default), the lowest region is shaded; otherwise, the next lowest region is shaded.
xlab	The label given to the sample proportion on the x-axis.
xmin	The minimum x-value to be graphed.
xmax	The maximum x-value to be graphed.

xtic	Logical or a vector of numbers. If xtic is TRUE (default), then the numbers on the x-axis include the median and xshade. If xtic is TRUE, then the default numbers from plot are listed on the x-axis. If xtic is a vector of numbers, then these numbers are listed on the x-axis.
digits.prob	The number of significant digits listed in the probability.
digits.xtic	The number of significant digits listed on the x-axis.
main	The main title given for the graph.
col	A vector of size two, specifying the colors of the density curve and the shading, respectively.
lwd	The line width illustrating the discrete probabilities.
...	Optional arguments to be passed to the plot function (see par).

Details

When illustrating a left-sided p-value or any other left-sided probability, xshade should be a single number and set lower.tail=TRUE (default). When illustrating a right-sided p-value or any other right-sided probability, xshade should be a single number and set lower.tail=FALSE. When illustrating a two-sided p-value or any other two-sided probability, xshade should be a vector of two numbers and set lower.tail=TRUE (default). When illustrating the complement of a two-sided p-value or the complement of any other two-sided probability, xshade should be a vector of two numbers and set lower.tail=FALSE.

This function shadePhat can be executed directly or indirectly via shadeDist.

Note

This function shadePhat calls functions plot and curve.

Author(s)

Steven T. Garren, James Madison University, Harrisonburg, Virginia, USA

See Also

shadeDist

Examples

```
shadePhat( 0.3, 20, 0.4 )
shadePhat( 0.3, 20, 0.4, lower.tail = FALSE )
shadePhat( c(0.65, 0.75), 30, 0.7, col=c("purple","orange") )
shadePhat( c(0.65, 0.75), 30, 0.7, lower.tail = FALSE )
shadePhat( c(0.3, 0.4), 50, 0.35, xmin=0.1, xmax=0.6, col=c("blue","lightgreen") )
```

Index