

Package ‘CGPfunctions’

June 14, 2018

Title Powell Miscellaneous Functions for Teaching and Learning
Statistics

Version 0.4

Description Miscellaneous functions useful for teaching statistics as well as actually practicing the art. They typically are not “new” methods but rather wrappers around either base R or other packages. Currently contains: 'Plot2WayANOVA' which as the name implies conducts a 2 way ANOVA and plots the results using 'ggplot2'. 'neweta' which is a helper function that appends the results of a Type II eta squared calculation onto a classic ANOVA table. Mode which finds the modal value in a vector of data. 'SeeDist' which wraps around 'ggplot2' to provide visualizations of univariate data.

Depends R (>= 3.4.0)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports car, devtools, dplyr, ggplot2 (>= 2.2.1), ggrepel, methods,
rlang (>= 0.1.6), scales, tibble

Suggests lsr, knitr, productplots, rmarkdown

VignetteBuilder knitr

RoxygenNote 6.0.1

URL <https://github.com/ibecav/CGPfunctions>

BugReports <https://github.com/ibecav/CGPfunctions/issues>

NeedsCompilation no

Author Chuck Powell [aut, cre]

Maintainer Chuck Powell <ibecav@gmail.com>

Repository CRAN

Date/Publication 2018-06-14 13:41:44 UTC

R topics documented:

CGPfunctions	2
Mode	3
newcancer	3
neweta	4
newggslopegraph	5
OurConf	7
Plot2WayANOVA	8
PlotXTabs	9
SeeDist	10
Index	12

CGPfunctions	<i>CGPfunctions: A package of miscellaneous functions for teaching statistics.</i>
--------------	--

Description

A package that includes miscellaneous functions useful for teaching statistics as well as actually practicing the art. They typically are not “new” methods but rather wrappers around either base R or other packages.

Functions included

- [Plot2WayANOVA](#) which as the name implies conducts a 2 way ANOVA and plots the results using ‘ggplot2’
- [PlotXTabs](#) Plots cross tabulated variables using ‘ggplot2’
- [neweta](#) which is a helper function that appends the results of a Type II eta squared calculation onto a classic ANOVA table
- [Mode](#) which finds the modal value in a vector of data
- [SeeDist](#) which wraps around ggplot2 to provide visualizations of univariate data.
- [OurConf](#) which wraps around ggplot2 to provide visualizations of sampling confidence intervals.

Mode	<i>Derive the modal value(s) for a set of data</i>
------	--

Description

This function takes a vector and returns one or mode values that represent the mode point of the data

Usage

```
Mode(x)
```

Arguments

x a vector

Value

a vector containing one or more modal values for the input vector

Warning

Be careful the function does some basic error checking but the return to Mode(NA) is NA and a vector where the majority of entries are NA is also NA

Examples

```
Mode(sample(1:100,1000,replace=TRUE))  
Mode(mtcars$hp)  
Mode(iris$Sepal.Length)
```

newcancer	<i>Tufte dataset on cancer survival rates</i>
-----------	---

Description

A dataset containing cancer survival rates for different types of cancer over a 20 year period.

Usage

```
newcancer
```

Format

A data frame with 96 rows and 3 variables:

Year ordered factor for the 5, 10, 15 and 20 year survival rates

Type factor containing the name of the cancer type

Survival numeric for this data a whole number corresponding to the percent survival rate

Source

https://www.edwardtufte.com/bboard/q-and-a-fetch-msg?msg_id=0003nk

neweta

Calculates eta squared for an AOV model using the Type II method

Description

Takes an aov object and returns a standard AOV table with eta squared computed

Usage

```
neweta(MyAOV)
```

Arguments

MyAOV a valid aov object such as those produced by `aov(dv~iv1*iv2)`

Details

There are three ways to compute eta squared this function only uses Type II

Value

Returns a tibble containing the AOV output similar to `summary(aov(MyAOV))` but with eta squared computed and appended as an additional column

Author(s)

Chuck Powell

References

neweta function is a shortened and simplified version of Dani Navarro's [etaSquared](#)

See Also

[Plot2WayANOVA](#)

Examples

```
neweta(aov(mpg~am*cyl, mtcars))
```

```
newggslopegraph          Plot a Slopegraph a la Tufte using dplyr and ggplot2
```

Description

Creates a "slopegraph" as conceptualized by Edward Tufte. Slopegraphs are minimalist and efficient presentations of your data that can simultaneously convey the relative rankings, the actual numeric values, and the changes and directionality of the data over time. Takes a dataframe as input, with three named columns being used to draw the plot. Makes the required adjustments to the ggplot2 parameters and returns the plot.

Usage

```
newggslopegraph(dataframe, Times, Measurement, Grouping,
  Title = "No title given", SubTitle = "No subtitle given",
  Caption = "No caption given", XTextSize = 12, YTextSize = 3,
  TitleTextSize = 14, SubTitleTextSize = 10, CaptionTextSize = 8,
  LineThickness = 1, LineColor = "ByGroup", DataTextSize = 2.5)
```

Arguments

dataframe	a dataframe or an object that can be coerced to a dataframe. Basic error checking is performed, to include ensuring that the named columns exist in the dataframe. See the newcancer dataset for an example of how the dataframe should be organized.
Times	a column inside the dataframe that will be plotted on the x axis. Traditionally this is some measure of time. The function accepts a column of class ordered, factor or character. NOTE if your variable is currently a "date" class you must convert before using the function with <code>as.character(variablename)</code> .
Measurement	a column inside the dataframe that will be plotted on the y axis. Traditionally this is some measure such as a percentage. Currently the function accepts a column of type integer or numeric. The slopegraph will be most effective when the measurements are not too disparate.
Grouping	a column inside the dataframe that will be used to group and distinguish measurements.
Title	Optionally the title to be displayed. Title = NULL will remove it entirely. Title = "" will provide an empty title but retain the spacing.
SubTitle	Optionally the sub-title to be displayed. SubTitle = NULL will remove it entirely. SubTitle = "" will provide and empty title but retain the spacing.
Caption	Optionally the caption to be displayed. Caption = NULL will remove it entirely. Caption = "" will provide and empty title but retain the spacing.

X textSize	Optionally the font size for the X axis labels to be displayed. X textSize = 12 is the default must be a numeric. Note that X & Y axis text are on different scales
Y textSize	Optionally the font size for the Y axis labels to be displayed. Y textSize = 3 is the default must be a numeric. Note that X & Y axis text are on different scales
Title textSize	Optionally the font size for the Title to be displayed. Title textSize = 14 is the default must be a numeric.
SubTitle textSize	Optionally the font size for the SubTitle to be displayed. SubTitle textSize = 10 is the default must be a numeric.
Caption textSize	Optionally the font size for the Caption to be displayed. Caption textSize = 8 is the default must be a numeric.
Line Thickness	Optionally the thickness of the plotted lines. Line Thickness = 1 is the default must be a numeric.
Line Color	Optionally the color of the plotted lines. By default it will use the ggplot2 color palette for coloring by Grouping. The user may override with one valid color of their choice e.g. "black" (see colors() for choices) OR they may provide a vector of colors such as c("gray", "red", "green", "gray", "blue") OR a named vector like c("Green" = "gray", "Liberal" = "red", "NDP" = "green", "Others" = "gray", "PC" = "blue"). Any input must be character, and the length of a vector should equal the number of levels in Grouping. If the user does not provide enough colors they will be recycled.
Data textSize	Optionally the font size of the plotted data points. Data textSize = 2.5 is the default must be a numeric.

Value

a plot of type ggplot to the default plot device

Author(s)

Chuck Powell

References

Based on: Edward Tufte, Beautiful Evidence (2006), pages 174-176.

See Also

[newcancer](#)

Examples

```
newggslopegraph(newcancer, Year, Survival, Type)
newggslopegraph(newcancer, Year, Survival, Type, Title = "Estimates of Percent Survival Rates")
newggslopegraph(newcancer, Year, Survival, Type,
                 Title = "Estimates of Percent Survival Rates",
```

```

      LineColor = "black",
      LineThickness = 1,
      SubTitle = NULL,
      Caption = NULL)
newggslopegraph(newcancer, Year, Survival, Type,
  Title = "Estimates of Percent Survival Rates",
  SubTitle = "Based on: Edward Tufte, Beautiful Evidence, 174, 176.",
  Caption = "https://www.edwardtufte.com/bboard/q-and-a-fetch-msg?msg_id=0003nk",
  LineColor = c("black", "red", "grey"),
  LineThickness = .5)

```

OurConf

Plotting random samples of confidence intervals around the mean

Description

This function takes some parameters and simulates random samples and their confidence intervals

Usage

```
OurConf(samples = 100, n = 30, mu = 0, sigma = 1, conf.level = 0.95)
```

Arguments

<code>samples</code>	The number of times to draw random samples
<code>n</code>	The sample size we draw each time
<code>mu</code>	The population mean μ
<code>sigma</code>	The population standard deviation
<code>conf.level</code>	What confidence level to compute $1 - \alpha$ (significance level)

Value

A ggplot2 object

Author(s)

Chuck Powell

See Also

[qnorm](#), [rnorm](#), [CIsim](#)

Examples

```

OurConf(samples = 100, n = 30, mu = 0, sigma = 1, conf.level = 0.95)
OurConf(samples = 2, n = 5)
OurConf(samples = 25, n = 25, mu = 100, sigma = 20, conf.level = 0.99)

```

Plot2WayANOVA

Plot a 2 Way ANOVA using dplyr and ggplot2

Description

Takes a formula and a dataframe as input, conducts an analysis of variance using the base R `aov` command and produces the results (AOV summary table and table of means) to the console and as a plotted interaction graph (line or bar) using `ggplot2`. Also uses Brown-Forsythe test for homogeneity of variance.

Usage

```
Plot2WayANOVA(formula, dataframe = NULL, confidence=.95,
              plottype = "bar", PlotSave = FALSE)
```

Arguments

<code>formula</code>	a valid R formula with a numeric dependent (outcome) variable, and two independent (predictor) variables e.g. <code>mpg~am*vs</code> . The independent variables are forced to factors (with warning) if possible.
<code>dataframe</code>	a dataframe or an object that can be coerced to a dataframe
<code>confidence</code>	what confidence level for confidence intervals
<code>plottype</code>	bar or line (quoted)
<code>PlotSave</code>	a logical indicating whether the user wants to save the plot as a png file

Details

Details about how the function works in order of steps taken.

1. Some basic error checking to ensure a valid formula and dataframe. Only accepts fully crossed formula to check for interaction term
2. Ensure the dependent (outcome) variable is numeric and that the two independent (predictor) variables are or can be coerced to factors – user warned on the console
3. Remove missing cases – user warned on the console
4. Use `dplyr` to calculate a summarized table of means, sds, standard errors of the means, confidence intervals, and group sizes.
5. Use the `aov` function to execute an Analysis of Variance (ANOVA)
6. Use the `neweta` function to calculate eta squared values. If the design is unbalanced warn the user and use Type II sums of squares
7. Produce a standard ANOVA table with a column for eta-squared appended
8. Use the `leveneTest` for testing Homogeneity of Variance assumption with Brown-Forsythe
9. Use the `shapiro.test` for testing normality assumption with Shapiro-Wilk
10. Use `ggplot2` to plot an interaction plot of the type the user specified

Value

A list with 4 elements which is returned invisibly. The items are always sent to the console for display. The plot is always sent to the default plot device but for user convenience the function also returns a named list with the following items in case the user desires to save them or further process them. `$ANOVATable`, `$MeansTable`, `$BFTest`, and `$SWTest`.

Author(s)

Chuck Powell

See Also

[aov](#), [leveneTest](#), [neweta](#), [replications](#), [shapiro.test](#)

Examples

```
Plot2WayANOVA(mpg~am*cyl, mtcars, plottype = "line")
Plot2WayANOVA(mpg~am*vs, mtcars, confidence = .99)
```

 PlotXTabs

Plot a Cross Tabulation of two variables using dplyr and ggplot2

Description

Takes a dataframe and at least two variables as input, conducts a crosstabulation of the variables using `dplyr`. Removes NAs and then plots the results as one of three types of bar (column) graphs using `ggplot2`. The function accepts either bare variable names or column numbers as input (see examples for the possibilities)

Usage

```
PlotXTabs(dataframe, xwhich, ywhich, plottype = "side")
```

Arguments

<code>dataframe</code>	an object that is of class dataframe
<code>xwhich</code>	either a bare variable name that is valid in the dataframe or one or more column numbers. An attempt will be made to coerce the variable to a factor but odd plots will occur if you pass it a variable that is by rights continuous in nature.
<code>ywhich</code>	either a bare variable name that is valid in the dataframe or one or more column numbers that exist in the dataframe. An attempt will be made to coerce the variable to a factor but odd plots will occur if you pass it a variable that is by rights continuous in nature.
<code>plottype</code>	one of three options "side", "stack" or "percent"

Value

One or more ggplots to the default graphics device as well as advisory information in the console

Author(s)

Chuck Powell

See Also

[janitor](#)

Examples

```
PlotXTabs(mtcars, am, vs)
PlotXTabs(mtcars, am, vs, "stack")
PlotXTabs(mtcars, am, vs, "percent")
PlotXTabs(mtcars, am, 8, "side")
PlotXTabs(mtcars, 8, am, "stack")
PlotXTabs(mtcars, am, c(8,10), "percent")
PlotXTabs(mtcars, c(10,8), am)
PlotXTabs(mtcars, c(2,9), c(10,8), "misspelled")
## Not run:
PlotXTabs(happy,happy,sex) # baseline
PlotXTabs(happy,2,5,"stack") # same thing using column numbers
PlotXTabs(happy, 2, c(5:9), plottype = "percent") # multiple columns RHS
PlotXTabs(happy, c(2,5), 9, plottype = "side") # multiple columns LHS
PlotXTabs(happy, c(2,5), c(6:9), plottype = "percent")
PlotXTabs(happy, happy, c(6,7,9), plottype = "percent")
PlotXTabs(happy, c(6,7,9), happy, plottype = "percent")

## End(Not run)
```

SeeDist

See The Distribution

Description

This function takes a vector of numeric data and returns one or more ggplot2 plots that help you visualize the data

Usage

```
SeeDist(qqq, numbins = 0, whatvar = "Unspecified", whatplots = c("d", "b",
  "h"))
```

Arguments

qqq	the data to be visualized must be numeric.
numbins	the number of bins to use for any plots that bin. If nothing is specified the function will calculate a rational number using Freedman-Diaconis via the <code>nclass.FD</code> function
whatvar	additional contextual information about the variable as a string such as "Miles Per Gallon"
whatplots	what type of plots? The default is <code>whatplots = c("d","b","h")</code> for a density, a boxplot, and a histogram

Value

from 1 to 3 plots depending on what the user specifies as well as a base R summary printed to the console

Warning

If the data has more than 3 modal values only the first three of them are plotted. The rest are ignored and the user is warned on the console.

Missing values are removed with a warning to the user

Author(s)

Chuck Powell

See Also

[nclass](#)

Examples

```
SeeDist(rnorm(100, mean=100, sd=20), numbins = 15, whatvar = "A Random Sample")
SeeDist(mtcars$hp, whatvar = "Horsepower", whatplots = c("d","b"))
SeeDist(iris$Sepal.Length, whatvar = "Sepal Length", whatplots = "d")
```

Index

*Topic **datasets**

newcancer, [3](#)

aov, [9](#)

CGPfunctions, [2](#)

CGPfunctions-package (CGPfunctions), [2](#)

CIsim, [7](#)

etaSquared, [4](#)

janitor, [10](#)

leveneTest, [9](#)

Mode, [2, 3](#)

nclass, [11](#)

newcancer, [3, 5, 6](#)

neweta, [2, 4, 8, 9](#)

newggslopegraph, [5](#)

OurConf, [2, 7](#)

Plot2WayANOVA, [2, 4, 8](#)

PlotXTabs, [2, 9](#)

qnorm, [7](#)

replications, [9](#)

rnorm, [7](#)

SeeDist, [2, 10](#)

shapiro.test, [9](#)