

Package ‘radiant.basics’

April 21, 2018

Type Package

Title Basics Menu for Radiant: Business Analytics using R and Shiny

Version 0.9.3.0

Date 2018-4-18

Description The Radiant Basics menu includes interfaces for probability calculation, central limit theorem simulation, comparing means and proportions, goodness-of-fit testing, cross-tabs, and correlation. The application extends the functionality in radiant.data.

Depends R (>= 3.4.0),
radiant.data (>= 0.9.3.0)

Imports ggplot2 (>= 2.2.1),
gridExtra (>= 2.0.0),
scales (>= 0.4.0),
dplyr (>= 0.7.4),
tidyr (>= 0.8.0),
magrittr (>= 1.5),
shiny (>= 1.0.5),
psych (>= 1.8.3.3),
import (>= 1.1.0),
rstudioapi (>= 0.7),
methods

Suggests testthat (>= 2.0.0)

URL <https://github.com/radiant-rstats/radiant.basics>, <https://radiant-rstats.github.io/docs>

BugReports <https://github.com/radiant-rstats/radiant.basics/issues>

License AGPL-3 | file LICENSE

LazyData true

Encoding UTF-8

RoxygenNote 6.0.1

R topics documented:

clt	3
compare_means	4
compare_props	5

consider	6
correlation	6
cross_tabs	7
demand_uk	8
goodness	8
newspaper	9
plot.clt	9
plot.compare_means	10
plot.compare_props	11
plot.correlation	12
plot.cross_tabs	12
plot.goodness	13
plot.prob_binom	14
plot.prob_chisq	15
plot.prob_disc	15
plot.prob_expo	16
plot.prob_fdist	16
plot.prob_lnorm	17
plot.prob_norm	17
plot.prob_pois	18
plot.prob_tdist	18
plot.prob_unif	19
plot.single_mean	19
plot.single_prop	20
print.correlation	21
prob_binom	21
prob_chisq	22
prob_disc	22
prob_expo	23
prob_fdist	23
prob_lnorm	24
prob_norm	24
prob_pois	25
prob_tdist	25
prob_unif	26
radiant.basics	27
radiant.basics_viewer	27
radiant.basics_window	28
salary	28
single_mean	29
single_prop	30
summary.compare_means	31
summary.compare_props	31
summary.correlation	32
summary.cross_tabs	33
summary.goodness	34
summary.prob_binom	35
summary.prob_chisq	35
summary.prob_disc	36
summary.prob_expo	36
summary.prob_fdist	37
summary.prob_lnorm	37

<i>clt</i>	3
summary.prob_norm	38
summary.prob_pois	38
summary.prob_tdist	39
summary.prob_unif	39
summary.single_mean	40
summary.single_prop	40
Index	42

<i>clt</i>	<i>Central Limit Theorem simulation</i>
------------	---

Description

Central Limit Theorem simulation

Usage

```
clt(dist, n = 100, m = 100, norm_mean = 0, norm_sd = 1,
    binom_size = 10, binom_prob = 0.2, unif_min = 0, unif_max = 1,
    expo_rate = 1)
```

Arguments

<code>dist</code>	Distribution to simulate
<code>n</code>	Sample size
<code>m</code>	Number of samples
<code>norm_mean</code>	Mean for the normal distribution
<code>norm_sd</code>	Standard deviation for the normal distribution
<code>binom_size</code>	Size for the binomial distribution
<code>binom_prob</code>	Probability for the binomial distribution
<code>unif_min</code>	Minimum for the uniform distribution
<code>unif_max</code>	Maximum for the uniform distribution
<code>expo_rate</code>	Rate for the exponential distribution

Details

See <https://radiant-rstats.github.io/docs/basics/clt.html> for an example in Radiant

compare_means	<i>Compare means for two or more variables</i>
---------------	--

Description

Compare means for two or more variables

Usage

```
compare_means(dataset, var1, var2, samples = "independent",
  alternative = "two.sided", conf_lev = 0.95, comb = "",
  adjust = "none", test = "t", data_filter = "")
```

Arguments

dataset	Dataset
var1	A numeric variable or factor selected for comparison
var2	One or more numeric variables for comparison. If var1 is a factor only one variable can be selected and the mean of this variable is compared across (factor) levels of var1
samples	Are samples independent ("independent") or not ("paired")
alternative	The alternative hypothesis ("two.sided", "greater" or "less")
conf_lev	Span of the confidence interval
comb	Combinations to evaluate
adjust	Adjustment for multiple comparisons ("none" or "bonf" for Bonferroni)
test	t-test ("t") or Wilcox ("wilcox")
data_filter	Expression entered in, e.g., Data > View to filter the dataset in Radiant. The expression should be a string (e.g., "price > 10000")

Details

See https://radiant-rstats.github.io/docs/basics/compare_means.html for an example in Radiant

Value

A list of all variables defined in the function as an object of class `compare_means`

See Also

[summary.compare_means](#) to summarize results

[plot.compare_means](#) to plot results

Examples

```
result <- compare_means(diamonds, "cut", "price")
result <- diamonds %>% compare_means("cut", "price", comb = c("Fair:Good", "Premium:Ideal"))
```

compare_props	<i>Compare proportions across groups</i>
---------------	--

Description

Compare proportions across groups

Usage

```
compare_props(dataset, var1, var2, levs = "", alternative = "two.sided",  
  conf_lev = 0.95, comb = "", adjust = "none", data_filter = "")
```

Arguments

dataset	Dataset
var1	A grouping variable to split the data for comparisons
var2	The variable to calculate proportions for
levs	The factor level selected for the proportion comparison
alternative	The alternative hypothesis ("two.sided", "greater" or "less")
conf_lev	Span of the confidence interval
comb	Combinations to evaluate
adjust	Adjustment for multiple comparisons ("none" or "bonf" for Bonferroni)
data_filter	Expression entered in, e.g., Data > View to filter the dataset in Radiant. The expression should be a string (e.g., "price > 10000")

Details

See https://radiant-rstats.github.io/docs/basics/compare_props.html for an example in Radiant

Value

A list of all variables defined in the function as an object of class `compare_props`

See Also

[summary.compare_props](#) to summarize results

[plot.compare_props](#) to plot results

Examples

```
result <- compare_props(titanic, "pclass", "survived")  
result <- titanic %>% compare_props("pclass", "survived")
```

consider	<i>Car brand consideration</i>
----------	--------------------------------

Description

Car brand consideration

Usage

```
data(consider)
```

Format

A data frame with 1000 rows and 2 variables

Details

Survey data of consumer purchase intentions. Description provided in `attr(consider,"description")`

correlation	<i>Calculate correlations for two or more variables</i>
-------------	---

Description

Calculate correlations for two or more variables

Usage

```
correlation(dataset, vars = "", method = "pearson", data_filter = "")
```

Arguments

dataset	Dataset
vars	Variables to include in the analysis. Default is all but character and factor variables with more than two unique values are removed
method	Type of correlations to calculate. Options are "pearson", "spearman", and "kendall". "pearson" is the default
data_filter	Expression entered in, e.g., Data > View to filter the dataset in Radiant. The expression should be a string (e.g., "price > 10000")

Details

See <https://radiant-rstats.github.io/docs/basics/correlation.html> for an example in Radiant

Value

A list with all variables defined in the function as an object of class `compare_means`

See Also

[summary.correlation](#) to summarize results

[plot.correlation](#) to plot results

Examples

```
result <- correlation(diamonds, c("price","carat"))
result <- correlation(diamonds, c("price","carat","table"))
result <- correlation(diamonds, "price:carat")
result <- diamonds %>% correlation("price:carat")
```

cross_tabs

*Evaluate associations between categorical variables***Description**

Evaluate associations between categorical variables

Usage

```
cross_tabs(dataset, var1, var2, tab = NULL, data_filter = "")
```

Arguments

dataset	Dataset (i.e., a data.frame or table)
var1	A categorical variable
var2	A categorical variable
tab	Table with frequencies as alternative to dataset
data_filter	Expression entered in, e.g., Data > View to filter the dataset in Radiant. The expression should be a string (e.g., "price > 10000")

Details

See https://radiant-rstats.github.io/docs/basics/cross_tabs.html for an example in Radiant

Value

A list of all variables used in cross_tabs as an object of class cross_tabs

See Also

[summary.cross_tabs](#) to summarize results

[plot.cross_tabs](#) to plot results

Examples

```
result <- cross_tabs(newspaper, "Income", "Newspaper")
result <- newspaper %>% cross_tabs("Income", "Newspaper")
result <- table(select(newspaper, Income, Newspaper)) %>% cross_tabs(tab = .)
```

demand_uk	<i>Demand in the UK</i>
-----------	-------------------------

Description

Demand in the UK

Usage

```
data(demand_uk)
```

Format

A data frame with 1000 rows and 2 variables

Details

Survey data of consumer purchase intentions. Description provided in attr(demand_uk,"description")

goodness	<i>Evaluate if sample data for a categorical variable is consistent with a hypothesized distribution</i>
----------	--

Description

Evaluate if sample data for a categorical variable is consistent with a hypothesized distribution

Usage

```
goodness(dataset, var, p = NULL, tab = NULL, data_filter = "")
```

Arguments

dataset	Dataset
var	A categorical variable
p	Hypothesized distribution as a number, fraction, or numeric vector. If unspecified, defaults to an even distribution
tab	Table with frequencies as alternative to dataset
data_filter	Expression entered in, e.g., Data > View to filter the dataset in Radiant. The expression should be a string (e.g., "price > 10000")

Details

See <https://radiant-rstats.github.io/docs/basics/goodness.html> for an example in Radiant

Value

A list of all variables used in goodness as an object of class goodness

See Also

[summary.goodness](#) to summarize results

[plot.goodness](#) to plot results

Examples

```
result <- goodness(newspaper, "Income")
result <- table(select(newspaper, Income)) %>% goodness(tab = .)
```

newspaper	<i>Newspaper readership</i>
-----------	-----------------------------

Description

Newspaper readership

Usage

```
data(newspaper)
```

Format

A data frame with 580 rows and 2 variables

Details

Newspaper readership data for 580 consumers. Description provided in `attr(newspaper, "description")`

plot.clt	<i>Plot method for the Central Limit Theorem function</i>
----------	---

Description

Plot method for the Central Limit Theorem function

Usage

```
## S3 method for class 'clt'
plot(x, stat = "sum", bins = 15, ...)
```

Arguments

x	Return value from clt
stat	Statistic to use (sum or mean)
bins	Number of bins to use
...	further arguments passed to or from other methods

Details

See <https://radiant-rstats.github.io/docs/basics/clt.html> for an example in Radiant

plot.compare_means	<i>Plot method for the compare_means function</i>
--------------------	---

Description

Plot method for the compare_means function

Usage

```
## S3 method for class 'compare_means'  
plot(x, plots = "scatter", shiny = FALSE,  
      custom = FALSE, ...)
```

Arguments

x	Return value from compare_means
plots	One or more plots ("bar", "density", "box", or "scatter")
shiny	Did the function call originate inside a shiny app
custom	Logical (TRUE, FALSE) to indicate if ggplot object (or list of ggplot objects) should be returned. This option can be used to customize plots (e.g., add a title, change x and y labels, etc.). See examples and http://docs.ggplot2.org/ for options.
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/compare_means.html for an example in Radiant

See Also

[compare_means](#) to calculate results
[summary.compare_means](#) to summarize results

Examples

```
result <- compare_means(diamonds, "cut", "price")  
plot(result, plots = c("bar", "density"))
```

plot.compare_props	<i>Plot method for the compare_props function</i>
--------------------	---

Description

Plot method for the compare_props function

Usage

```
## S3 method for class 'compare_props'  
plot(x, plots = "bar", shiny = FALSE,  
      custom = FALSE, ...)
```

Arguments

x	Return value from compare_props
plots	One or more plots of proportions ("bar" or "dodge")
shiny	Did the function call originate inside a shiny app
custom	Logical (TRUE, FALSE) to indicate if ggplot object (or list of ggplot objects) should be returned. This option can be used to customize plots (e.g., add a title, change x and y labels, etc.). See examples and http://docs.ggplot2.org/ for options.
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/compare_props.html for an example in Radiant

See Also

[compare_props](#) to calculate results
[summary.compare_props](#) to summarize results

Examples

```
result <- compare_props(titanic, "pclass", "survived")  
plot(result, plots = c("bar", "dodge"))
```

plot.correlation	<i>Plot method for the correlation function</i>
------------------	---

Description

Plot method for the correlation function

Usage

```
## S3 method for class 'correlation'
plot(x, nrobs = -1, jit = 0.3, ...)
```

Arguments

x	Return value from correlation
nrobs	Number of data points to show in scatter plots (-1 for all)
jit	Level of jittering to apply to scatter plot. Default is .3. Use 0 for no jittering
...	further arguments passed to or from other methods.

Details

See <https://radiant-rstats.github.io/docs/basics/correlation.html> for an example in Radiant

See Also

[correlation](#) to calculate results
[summary.correlation](#) to summarize results

Examples

```
result <- correlation(diamonds, c("price", "carat", "table"))
plot(result)
diamonds %>% correlation("price:carat") %>% plot()
```

plot.cross_tabs	<i>Plot method for the cross_tabs function</i>
-----------------	--

Description

Plot method for the cross_tabs function

Usage

```
## S3 method for class 'cross_tabs'
plot(x, check = "", shiny = FALSE, custom = FALSE,
     ...)
```

Arguments

x	Return value from cross_tabs
check	Show plots for variables var1 and var2. "observed" for the observed frequencies table, "expected" for the expected frequencies table (i.e., frequencies that would be expected if the null hypothesis holds), "chi_sq" for the contribution to the overall chi-squared statistic for each cell (i.e., $(o - e)^2 / e$), "dev_std" for the standardized differences between the observed and expected frequencies (i.e., $(o - e) / \sqrt{e}$), and "row_perc", "col_perc", and "perc" for row, column, and table percentages respectively
shiny	Did the function call originate inside a shiny app
custom	Logical (TRUE, FALSE) to indicate if ggplot object (or list of ggplot objects) should be returned. This option can be used to customize plots (e.g., add a title, change x and y labels, etc.). See examples and http://docs.ggplot2.org/ for options.
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/cross_tabs.html for an example in Radiant

See Also

[cross_tabs](#) to calculate results

[summary.cross_tabs](#) to summarize results

Examples

```
result <- cross_tabs(newspaper, "Income", "Newspaper")
plot(result, check = c("observed", "expected", "chi_sq"))
newspaper %>% cross_tabs("Income", "Newspaper") %>% plot(c("observed", "expected"))
```

plot.goodness

Plot method for the goodness function

Description

Plot method for the goodness function

Usage

```
## S3 method for class 'goodness'
plot(x, check = "", fillcol = "blue", shiny = FALSE,
     custom = FALSE, ...)
```

Arguments

x	Return value from goodness
check	Show plots for variable var. "observed" for the observed frequencies table, "expected" for the expected frequencies table (i.e., frequencies that would be expected if the null hypothesis holds), "chi_sq" for the contribution to the overall chi-squared statistic for each cell (i.e., $(o - e)^2 / e$), and "dev_std" for the standardized differences between the observed and expected frequencies (i.e., $(o - e) / \sqrt{e}$)
fillcol	Color used for bar plots
shiny	Did the function call originate inside a shiny app
custom	Logical (TRUE, FALSE) to indicate if ggplot object (or list of ggplot objects) should be returned. This option can be used to customize plots (e.g., add a title, change x and y labels, etc.). See examples and http://docs.ggplot2.org/ for options.
...	further arguments passed to or from other methods

Details

See <https://radiant-rstats.github.io/docs/basics/goodness> for an example in Radiant

See Also

[goodness](#) to calculate results
[summary.goodness](#) to summarize results

Examples

```
result <- goodness(newspaper, "Income")
plot(result, check = c("observed", "expected", "chi_sq"))
newspaper %>% goodness("Income") %>% plot(c("observed", "expected"))
```

plot.prob_binom

Plot method for the probability calculator function (binomial)

Description

Plot method for the probability calculator function (binomial)

Usage

```
## S3 method for class 'prob_binom'
plot(x, type = "values", ...)
```

Arguments

x	Return value from prob_binom
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

plot.prob_chisq	<i>Plot method for the probability calculator (Chi-squared distribution)</i>
-----------------	--

Description

Plot method for the probability calculator (Chi-squared distribution)

Usage

```
## S3 method for class 'prob_chisq'
plot(x, type = "values", ...)
```

Arguments

x	Return value from prob_chisq
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

plot.prob_disc	<i>Plot method for the probability calculator function (discrete)</i>
----------------	---

Description

Plot method for the probability calculator function (discrete)

Usage

```
## S3 method for class 'prob_disc'
plot(x, type = "values", ...)
```

Arguments

x	Return value from prob_disc
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

Examples

```
result <- prob_disc(v = "5 6 7 8 9 10 11 ", p = ".1 .2 .3 .15 .1 .1 .05", pub = 0.95)
plot(result, type = "probs")
```

plot.prob_expo	<i>Plot method for the probability calculator (Exponential distribution)</i>
----------------	--

Description

Plot method for the probability calculator (Exponential distribution)

Usage

```
## S3 method for class 'prob_expo'
plot(x, type = "values", ...)
```

Arguments

x	Return value from prob_expo
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

plot.prob_fdist	<i>Plot method for the probability calculator (F-distribution)</i>
-----------------	--

Description

Plot method for the probability calculator (F-distribution)

Usage

```
## S3 method for class 'prob_fdist'
plot(x, type = "values", ...)
```

Arguments

x	Return value from prob_fdist
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

plot.prob_lnorm	<i>Plot method for the probability calculator (log normal)</i>
-----------------	--

Description

Plot method for the probability calculator (log normal)

Usage

```
## S3 method for class 'prob_lnorm'  
plot(x, type = "values", ...)
```

Arguments

x	Return value from prob_norm
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radiant

plot.prob_norm	<i>Plot method for the probability calculator (normal)</i>
----------------	--

Description

Plot method for the probability calculator (normal)

Usage

```
## S3 method for class 'prob_norm'  
plot(x, type = "values", ...)
```

Arguments

x	Return value from prob_norm
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radiant

plot.prob_pois	<i>Plot method for the probability calculator function (Poisson distribution)</i>
----------------	---

Description

Plot method for the probability calculator function (Poisson distribution)

Usage

```
## S3 method for class 'prob_pois'
plot(x, type = "values", ...)
```

Arguments

x	Return value from prob_pois
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radiant

plot.prob_tdist	<i>Plot method for the probability calculator (t-distribution)</i>
-----------------	--

Description

Plot method for the probability calculator (t-distribution)

Usage

```
## S3 method for class 'prob_tdist'
plot(x, type = "values", ...)
```

Arguments

x	Return value from prob_tdist
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radiant

plot.prob_unif	<i>Plot method for the probability calculator (uniform)</i>
----------------	---

Description

Plot method for the probability calculator (uniform)

Usage

```
## S3 method for class 'prob_unif'
plot(x, type = "values", ...)
```

Arguments

x	Return value from prob_unif
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radiant

plot.single_mean	<i>Plot method for the single_mean function</i>
------------------	---

Description

Plot method for the single_mean function

Usage

```
## S3 method for class 'single_mean'
plot(x, plots = "hist", shiny = FALSE,
     custom = FALSE, ...)
```

Arguments

x	Return value from single_mean
plots	Plots to generate. "hist" shows a histogram of the data along with vertical lines that indicate the sample mean and the confidence interval. "simulate" shows the location of the sample mean and the comparison value (comp_value). Simulation is used to demonstrate the sampling variability in the data under the null-hypothesis
shiny	Did the function call originate inside a shiny app
custom	Logical (TRUE, FALSE) to indicate if ggplot object (or list of ggplot objects) should be returned. This option can be used to customize plots (e.g., add a title, change x and y labels, etc.). See examples and http://docs.ggplot2.org/ for options.
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/single_mean.html for an example in Radiant

See Also

[single_mean](#) to generate the result
[summary.single_mean](#) to summarize results

Examples

```
result <- single_mean(diamonds, "price", comp_value = 3500)
plot(result, plots = c("hist", "simulate"))
```

plot.single_prop	<i>Plot method for the single_prop function</i>
------------------	---

Description

Plot method for the single_prop function

Usage

```
## S3 method for class 'single_prop'
plot(x, plots = "bar", shiny = FALSE,
     custom = FALSE, ...)
```

Arguments

x	Return value from single_prop
plots	Plots to generate. "bar" shows a bar chart of the data. The "simulate" chart shows the location of the sample proportion and the comparison value (comp_value). Simulation is used to demonstrate the sampling variability in the data under the null-hypothesis
shiny	Did the function call originate inside a shiny app
custom	Logical (TRUE, FALSE) to indicate if ggplot object (or list of ggplot objects) should be returned. This option can be used to customize plots (e.g., add a title, change x and y labels, etc.). See examples and http://docs.ggplot2.org/ for options.
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/single_prop.html for an example in Radiant

See Also

[single_prop](#) to generate the result
[summary.single_prop](#) to summarize the results

Examples

```
result <- single_prop(diamonds,"clarity", lev = "IF", comp_value = 0.05)
plot(result, plots = c("bar", "simulate"))
result <- single_prop(titanic,"pclass", lev = "1st")
plot(result, plots = c("bar", "simulate"))
```

print.correlation	<i>Print method for radiant.basics::correlation</i>
-------------------	---

Description

Print method for radiant.basics::correlation

Usage

```
## S3 method for class 'correlation'
print(x, ...)
```

Arguments

x	Return value from correlation
...	further arguments passed to or from other methods

prob_binom	<i>Probability calculator for the binomial distribution (binomial)</i>
------------	--

Description

Probability calculator for the binomial distribution (binomial)

Usage

```
prob_binom(n, p, lb = NA, ub = NA, plb = NA, pub = NA, dec = 3)
```

Arguments

n	Number of trials
p	Probability
lb	Lower bound on the number of successes
ub	Upper bound on the number of successes
plb	Lower probability bound
pub	Upper probability bound
dec	Number of decimals to show

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radiant

prob_chisq	<i>Probability calculator for the chi-squared distribution</i>
------------	--

Description

Probability calculator for the chi-squared distribution

Usage

```
prob_chisq(df, lb = NA, ub = NA, plb = NA, pub = NA, dec = 3)
```

Arguments

df	Degrees of freedom
lb	Lower bound (default is 0)
ub	Upper bound (default is Inf)
plb	Lower probability bound
pub	Upper probability bound
dec	Number of decimals to show

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radiant

prob_disc	<i>Probability calculator for the discrete distribution (discrete)</i>
-----------	--

Description

Probability calculator for the discrete distribution (discrete)

Usage

```
prob_disc(v, p, lb = NA, ub = NA, plb = NA, pub = NA, dec = 3)
```

Arguments

v	Values
p	Probabilities
lb	Lower bound on the number of successes
ub	Upper bound on the number of successes
plb	Lower probability bound
pub	Upper probability bound
dec	Number of decimals to show

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radiant

prob_expo	<i>Probability calculator for the exponential distribution</i>
-----------	--

Description

Probability calculator for the exponential distribution

Usage

```
prob_expo(rate, lb = NA, ub = NA, plb = NA, pub = NA, dec = 3)
```

Arguments

rate	Rate
lb	Lower bound (default is 0)
ub	Upper bound (default is Inf)
plb	Lower probability bound
pub	Upper probability bound
dec	Number of decimals to show

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

prob_fdist	<i>Probability calculator for the F-distribution</i>
------------	--

Description

Probability calculator for the F-distribution

Usage

```
prob_fdist(df1, df2, lb = NA, ub = NA, plb = NA, pub = NA, dec = 3)
```

Arguments

df1	Degrees of freedom
df2	Degrees of freedom
lb	Lower bound (default is 0)
ub	Upper bound (default is Inf)
plb	Lower probability bound
pub	Upper probability bound
dec	Number of decimals to show

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

prob_lnorm	<i>Probability calculator for the log normal distribution</i>
------------	---

Description

Probability calculator for the log normal distribution

Usage

```
prob_lnorm(meanlog, sdlog, lb = NA, ub = NA, plb = NA, pub = NA,
  dec = 3)
```

Arguments

meanlog	Mean of the distribution on the log scale
sdlog	Standard deviation of the distribution on the log scale
lb	Lower bound (default is -Inf)
ub	Upper bound (default is Inf)
plb	Lower probability bound
pub	Upper probability bound
dec	Number of decimals to show

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radiant

prob_norm	<i>Probability calculator for the normal distribution</i>
-----------	---

Description

Probability calculator for the normal distribution

Usage

```
prob_norm(mean, stdev, lb = NA, ub = NA, plb = NA, pub = NA, dec = 3)
```

Arguments

mean	Mean
stdev	Standard deviation
lb	Lower bound (default is -Inf)
ub	Upper bound (default is Inf)
plb	Lower probability bound
pub	Upper probability bound
dec	Number of decimals to show

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radiant

prob_pois	<i>Probability calculator for the poisson distribution</i>
-----------	--

Description

Probability calculator for the poisson distribution

Usage

```
prob_pois(lambda, lb = NA, ub = NA, plb = NA, pub = NA, dec = 3)
```

Arguments

lambda	Rate
lb	Lower bound (default is 0)
ub	Upper bound (default is Inf)
plb	Lower probability bound
pub	Upper probability bound
dec	Number of decimals to show

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radiant

prob_tdist	<i>Probability calculator for the t distribution</i>
------------	--

Description

Probability calculator for the t distribution

Usage

```
prob_tdist(df, mean = 0, stdev = 1, lb = NA, ub = NA, plb = NA,  
pub = NA, dec = 3)
```

Arguments

df	Degrees of freedom
mean	Mean
stdev	Standard deviation
lb	Lower bound (default is -Inf)
ub	Upper bound (default is Inf)
plb	Lower probability bound
pub	Upper probability bound
dec	Number of decimals to show

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

prob_unif	<i>Probability calculator for the uniform distribution</i>
-----------	--

Description

Probability calculator for the uniform distribution

Usage

```
prob_unif(min, max, lb = NA, ub = NA, plb = NA, pub = NA, dec = 3)
```

Arguments

min	Minmum value
max	Maximum value
lb	Lower bound (default = 0)
ub	Upper bound (default = 1)
plb	Lower probability bound
pub	Upper probability bound
dec	Number of decimals to show

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

<code>radiant.basics</code>	<i><code>radiant.basics</code></i>
-----------------------------	------------------------------------

Description

`radiant.basics`

Launch `radiant.basics` in the default browser

Usage

```
radiant.basics()
```

Details

See <https://radiant-rstats.github.io/docs> for documentation and tutorials

Examples

```
## Not run:  
radiant.basics()  
  
## End(Not run)
```

<code>radiant.basics_viewer</code>	<i>Launch <code>radiant.basics</code> in the Rstudio viewer</i>
------------------------------------	---

Description

Launch `radiant.basics` in the Rstudio viewer

Usage

```
radiant.basics_viewer()
```

Details

See <https://radiant-rstats.github.io/docs> for documentation and tutorials

Examples

```
## Not run:  
radiant.basics_viewer()  
  
## End(Not run)
```

```
radiant.basics_window
```

Launch radiant.basics in an Rstudio window

Description

Launch radiant.basics in an Rstudio window

Usage

```
radiant.basics_window()
```

Details

See <https://radiant-rstats.github.io/docs> for documentation and tutorials

Examples

```
## Not run:  
radiant.basics_window()  
  
## End(Not run)
```

```
salary
```

Salaries for Professors

Description

Salaries for Professors

Usage

```
data(salary)
```

Format

A data frame with 397 rows and 6 variables

Details

2008-2009 nine-month salary for professors in a college in the US. Description provided in attr(salary,description")

`single_mean`*Compare a sample mean to a population mean*

Description

Compare a sample mean to a population mean

Usage

```
single_mean(dataset, var, comp_value = 0, alternative = "two.sided",  
             conf_lev = 0.95, data_filter = "")
```

Arguments

<code>dataset</code>	Dataset
<code>var</code>	The variable selected for the mean comparison
<code>comp_value</code>	Population value to compare to the sample mean
<code>alternative</code>	The alternative hypothesis ("two.sided", "greater", or "less")
<code>conf_lev</code>	Span for the confidence interval
<code>data_filter</code>	Expression entered in, e.g., Data > View to filter the dataset in Radiant. The expression should be a string (e.g., "price > 10000")

Details

See https://radiant-rstats.github.io/docs/basics/single_mean.html for an example in Radiant

Value

A list of variables defined in `single_mean` as an object of class `single_mean`

See Also

[summary.single_mean](#) to summarize results

[plot.single_mean](#) to plot results

Examples

```
single_mean(diamonds, "price")
```

`single_prop`*Compare a sample proportion to a population proportion*

Description

Compare a sample proportion to a population proportion

Usage

```
single_prop(dataset, var, lev = "", comp_value = 0.5,  
            alternative = "two.sided", conf_lev = 0.95, data_filter = "")
```

Arguments

<code>dataset</code>	Dataset
<code>var</code>	The variable selected for the proportion comparison
<code>lev</code>	The factor level selected for the proportion comparison
<code>comp_value</code>	Population value to compare to the sample proportion
<code>alternative</code>	The alternative hypothesis ("two.sided", "greater", or "less")
<code>conf_lev</code>	Span of the confidence interval
<code>data_filter</code>	Expression entered in, e.g., Data > View to filter the dataset in Radiant. The expression should be a string (e.g., "price > 10000")

Details

See https://radiant-rstats.github.io/docs/basics/single_prop.html for an example in Radiant

Value

A list of variables used in `single_prop` as an object of class `single_prop`

See Also

[summary.single_prop](#) to summarize the results

[plot.single_prop](#) to plot the results

Examples

```
result <- single_prop(diamonds, "cut")  
result <- single_prop(diamonds, "clarity", lev = "IF", comp_value = 0.05)
```

summary.compare_means *Summary method for the compare_means function*

Description

Summary method for the compare_means function

Usage

```
## S3 method for class 'compare_means'  
summary(object, show = FALSE, dec = 3, ...)
```

Arguments

object	Return value from compare_means
show	Show additional output (i.e., t.value, df, and confidence interval)
dec	Number of decimals to show
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/compare_means.html for an example in Radiant

See Also

[compare_means](#) to calculate results
[plot.compare_means](#) to plot results

Examples

```
result <- compare_means(diamonds, "cut", "price")  
summary(result)  
compare_means(diamonds, "price", "carat") %>% summary()
```

summary.compare_props *Summary method for the compare_props function*

Description

Summary method for the compare_props function

Usage

```
## S3 method for class 'compare_props'  
summary(object, show = FALSE, dec = 3, ...)
```

Arguments

object	Return value from compare_props
show	Show additional output (i.e., chisq.value, df, and confidence interval)
dec	Number of decimals to show
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/compare_props.html for an example in Radiant

See Also

[compare_props](#) to calculate results
[plot.compare_props](#) to plot results

Examples

```
result <- compare_props(titanic, "pclass", "survived")
summary(result)
titanic %>% compare_props("pclass", "survived") %>% summary
```

summary.correlation	<i>Summary method for the correlation function</i>
---------------------	--

Description

Summary method for the correlation function

Usage

```
## S3 method for class 'correlation'
summary(object, cutoff = 0, covar = FALSE, dec = 2,
  ...)
```

Arguments

object	Return value from correlation
cutoff	Show only correlations larger than the cutoff in absolute value. Default is a cutoff of 0
covar	Show the covariance matrix (default is FALSE)
dec	Number of decimals to show
...	further arguments passed to or from other methods.

Details

See <https://radiant-rstats.github.io/docs/basics/correlation.html> for an example in Radiant

See Also

[correlation](#) to calculate results
[plot.correlation](#) to plot results

Examples

```
result <- correlation(diamonds, c("price", "carat", "table"))
summary(result, cutoff = .3)
diamonds %>% correlation("price:carat") %>% summary
```

summary.cross_tabs	<i>Summary method for the cross_tabs function</i>
--------------------	---

Description

Summary method for the cross_tabs function

Usage

```
## S3 method for class 'cross_tabs'
summary(object, check = "", dec = 2, ...)
```

Arguments

object	Return value from cross_tabs
check	Show table(s) for variables var1 and var2. "observed" for the observed frequencies table, "expected" for the expected frequencies table (i.e., frequencies that would be expected if the null hypothesis holds), "chi_sq" for the contribution to the overall chi-squared statistic for each cell (i.e., $(o - e)^2 / e$), "dev_std" for the standardized differences between the observed and expected frequencies (i.e., $(o - e) / \sqrt{e}$), and "dev_perc" for the percentage difference between the observed and expected frequencies (i.e., $(o - e) / e$)
dec	Number of decimals to show
...	further arguments passed to or from other methods.

Details

See https://radiant-rstats.github.io/docs/basics/cross_tabs.html for an example in Radiant

See Also

[cross_tabs](#) to calculate results
[plot.cross_tabs](#) to plot results

Examples

```
result <- cross_tabs(newspaper, "Income", "Newspaper")
summary(result, check = c("observed", "expected", "chi_sq"))
newspaper %>% cross_tabs("Income", "Newspaper") %>% summary("observed")
```

summary.goodness	<i>Summary method for the goodness function</i>
------------------	---

Description

Summary method for the goodness function

Usage

```
## S3 method for class 'goodness'
summary(object, check = "", dec = 2, ...)
```

Arguments

object	Return value from goodness
check	Show table(s) for the selected variable (var). "observed" for the observed frequencies table, "expected" for the expected frequencies table (i.e., frequencies that would be expected if the null hypothesis holds), "chi_sq" for the contribution to the overall chi-squared statistic for each cell (i.e., $(o - e)^2 / e$), "dev_std" for the standardized differences between the observed and expected frequencies (i.e., $(o - e) / \sqrt{e}$), and "dev_perc" for the percentage difference between the observed and expected frequencies (i.e., $(o - e) / e$)
dec	Number of decimals to show
...	further arguments passed to or from other methods.

Details

See <https://radiant-rstats.github.io/docs/basics/goodness> for an example in Radiant

See Also

[goodness](#) to calculate results

[plot.goodness](#) to plot results

Examples

```
result <- goodness(newspaper, "Income", c(.3, .7))
summary(result, check = c("observed", "expected", "chi_sq"))
newspaper %>% goodness("Income", "1/3 2/3") %>% summary("observed")
```

summary.prob_binom	<i>Summary method for the probability calculator function</i>
--------------------	---

Description

Summary method for the probability calculator function

Usage

```
## S3 method for class 'prob_binom'  
summary(object, type = "values", ...)
```

Arguments

object	Return value from prob_binom
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

summary.prob_chisq	<i>Summary method for the probability calculator function (Chi-squared distribution)</i>
--------------------	--

Description

Summary method for the probability calculator function (Chi-squared distribution)

Usage

```
## S3 method for class 'prob_chisq'  
summary(object, type = "values", ...)
```

Arguments

object	Return value from prob_chisq
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

summary.prob_disc	<i>Summary method for the probability calculator function (discrete)</i>
-------------------	--

Description

Summary method for the probability calculator function (discrete)

Usage

```
## S3 method for class 'prob_disc'
summary(object, type = "values", ...)
```

Arguments

object	Return value from prob_disc
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radiant

Examples

```
result <- prob_disc(v = "5 6 7 8 9 10 11 ", p = ".1 .2 .3 .15 .1 .1 .05", pub = 0.95)
summary(result, type = "probs")
```

summary.prob_expo	<i>Summary method for the probability calculator function (Exponential distribution)</i>
-------------------	--

Description

Summary method for the probability calculator function (Exponential distribution)

Usage

```
## S3 method for class 'prob_expo'
summary(object, type = "values", ...)
```

Arguments

object	Return value from prob_expo
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radiant

summary.prob_fdist	<i>Summary method for the probability calculator function (F-distribution)</i>
--------------------	--

Description

Summary method for the probability calculator function (F-distribution)

Usage

```
## S3 method for class 'prob_fdist'  
summary(object, type = "values", ...)
```

Arguments

object	Return value from prob_fdist
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

summary.prob_lnorm	<i>Summary method for the probability calculator function (log normal)</i>
--------------------	--

Description

Summary method for the probability calculator function (log normal)

Usage

```
## S3 method for class 'prob_lnorm'  
summary(object, type = "values", ...)
```

Arguments

object	Return value from prob_norm
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

summary.prob_norm	<i>Summary method for the probability calculator function (normal)</i>
-------------------	--

Description

Summary method for the probability calculator function (normal)

Usage

```
## S3 method for class 'prob_norm'  
summary(object, type = "values", ...)
```

Arguments

object	Return value from prob_norm
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

summary.prob_pois	<i>Summary method for the probability calculator function (Poisson distribution)</i>
-------------------	--

Description

Summary method for the probability calculator function (Poisson distribution)

Usage

```
## S3 method for class 'prob_pois'  
summary(object, type = "values", ...)
```

Arguments

object	Return value from prob_pois
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

summary.prob_tdist	<i>Summary method for the probability calculator function (t-distribution)</i>
--------------------	--

Description

Summary method for the probability calculator function (t-distribution)

Usage

```
## S3 method for class 'prob_tdist'  
summary(object, type = "values", ...)
```

Arguments

object	Return value from prob_tdist
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

summary.prob_unif	<i>Summary method for the probability calculator function (uniform)</i>
-------------------	---

Description

Summary method for the probability calculator function (uniform)

Usage

```
## S3 method for class 'prob_unif'  
summary(object, type = "values", ...)
```

Arguments

object	Return value from prob_unif
type	Probabilities or values
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/prob_calc.html for an example in Radian

summary.single_mean	<i>Summary method for the single_mean function</i>
---------------------	--

Description

Summary method for the single_mean function

Usage

```
## S3 method for class 'single_mean'  
summary(object, dec = 3, ...)
```

Arguments

object	Return value from single_mean
dec	Number of decimals to show
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/single_mean.html for an example in Radiant

See Also

[single_mean](#) to generate the results
[plot.single_mean](#) to plot results

Examples

```
result <- single_mean(diamonds, "price")  
summary(result)  
diamonds %>% single_mean("price") %>% summary
```

summary.single_prop	<i>Summary method for the single_prop function</i>
---------------------	--

Description

Summary method for the single_prop function

Usage

```
## S3 method for class 'single_prop'  
summary(object, dec = 3, ...)
```


Arguments

object	Return value from single_prop
dec	Number of decimals to show
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/basics/single_prop.html for an example in Radiant

See Also

[single_prop](#) to generate the results
[plot.single_prop](#) to plot the results

Examples

```
result <- single_prop(diamonds,"clarity", lev = "IF", comp_value = 0.05)
summary(result)
diamonds %>% single_prop("clarity", lev = "IF", comp_value = 0.05) %>% summary
```

Index

*Topic **datasets**

- consider, [6](#)
- demand_uk, [8](#)
- newspaper, [9](#)
- salary, [28](#)

- clt, [3](#), [9](#)
- compare_means, [4](#), [10](#), [31](#)
- compare_props, [5](#), [11](#), [32](#)
- consider, [6](#)
- correlation, [6](#), [12](#), [32](#), [33](#)
- cross_tabs, [7](#), [13](#), [33](#)

- demand_uk, [8](#)

- goodness, [8](#), [14](#), [34](#)

- newspaper, [9](#)

- plot.clt, [9](#)
- plot.compare_means, [4](#), [10](#), [31](#)
- plot.compare_props, [5](#), [11](#), [32](#)
- plot.correlation, [7](#), [12](#), [33](#)
- plot.cross_tabs, [7](#), [12](#), [33](#)
- plot.goodness, [9](#), [13](#), [34](#)
- plot.prob_binom, [14](#)
- plot.prob_chisq, [15](#)
- plot.prob_disc, [15](#)
- plot.prob_expo, [16](#)
- plot.prob_fdist, [16](#)
- plot.prob_lnorm, [17](#)
- plot.prob_norm, [17](#)
- plot.prob_pois, [18](#)
- plot.prob_tdist, [18](#)
- plot.prob_unif, [19](#)
- plot.single_mean, [19](#), [29](#), [40](#)
- plot.single_prop, [20](#), [30](#), [41](#)
- print.correlation, [21](#)
- prob_binom, [14](#), [21](#), [35](#)
- prob_chisq, [15](#), [22](#), [35](#)
- prob_disc, [15](#), [22](#), [36](#)
- prob_expo, [16](#), [23](#), [36](#)
- prob_fdist, [16](#), [23](#), [37](#)
- prob_lnorm, [24](#)
- prob_norm, [17](#), [24](#), [37](#), [38](#)

- prob_pois, [18](#), [25](#), [38](#)
- prob_tdist, [18](#), [25](#), [39](#)
- prob_unif, [19](#), [26](#), [39](#)

- radiant.basics, [27](#)
- radiant.basics-package
 - (radiant.basics), [27](#)
- radiant.basics_viewer, [27](#)
- radiant.basics_window, [28](#)

- salary, [28](#)
- single_mean, [19](#), [20](#), [29](#), [40](#)
- single_prop, [20](#), [30](#), [41](#)
- summary.compare_means, [4](#), [10](#), [31](#)
- summary.compare_props, [5](#), [11](#), [31](#)
- summary.correlation, [7](#), [12](#), [32](#)
- summary.cross_tabs, [7](#), [13](#), [33](#)
- summary.goodness, [9](#), [14](#), [34](#)
- summary.prob_binom, [35](#)
- summary.prob_chisq, [35](#)
- summary.prob_disc, [36](#)
- summary.prob_expo, [36](#)
- summary.prob_fdist, [37](#)
- summary.prob_lnorm, [37](#)
- summary.prob_norm, [38](#)
- summary.prob_pois, [38](#)
- summary.prob_tdist, [39](#)
- summary.prob_unif, [39](#)
- summary.single_mean, [20](#), [29](#), [40](#)
- summary.single_prop, [20](#), [30](#), [40](#)