Package 'correlatio'

May 24, 2025

Type Package

Title Visualize Details Behind Pearson's Correlation Coefficient

Version 0.2.1

Description

Helps visualizing what is summarized in Pearson's correlation coefficient. That is, it visualizes its main constituent, namely the distances of the single values to their respective mean. The visualization thereby shows what the etymology of the word correlation contains: In pairwise combination, bringing back (see package Vignette for more details). I hope that the 'correlatio' package may benefit some people in understanding and critically evaluating what Pearson's correlation coefficient summarizes in a single number, i.e., to what degree and why Pearson's correlation coefficient may (or may not) be warranted as a measure of association.

License MIT + file LICENSE

Repository CRAN

URL https://github.com/mmiche/correlatio

Encoding UTF-8

Imports Rdpack, stats, ggplot2, tibble

RdMacros Rdpack

RoxygenNote 7.3.2

Depends R (>= 4.0.0)

Suggests knitr, rmarkdown

VignetteBuilder knitr

BugReports https://github.com/mmiche/correlatio/issues

NeedsCompilation no

Author Marcel Miché [aut, cre]

Maintainer Marcel Miché <marcel.miche.predictme@gmail.com>

Date/Publication 2025-05-24 15:50:02 UTC

Contents

---

correlatio	*Documentation of this correlatio package.*

Description

This R package can help visualizing what is summarized in Pearson's correlation coefficient.

All R packages that have been used in, as well as for developing, this correlatio package, are listed below. Thanks to the many R package developers!

References

Wickham H, Wickham H (2016). "Programming with ggplot2." *Ggplot2: elegant graphics for data analysis*, 241–253.

Müller K, Wickham H (2023). *tibble: Simple Data Frames*. R package version 3.2.1, https://CRAN.R-project.org/package=tibble.

R Core Team (2024). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/.

Wickham H, Hester J, Chang W, Bryan J (2021). *devtools: Tools to Make Developing R Packages Easier*. R package version 2.4.3, https://CRAN.R-project.org/package=devtools.

Wickham H, Bryan J (2021). *usethis: Automate Package and Project Setup*. R package version 2.0.1, https://CRAN.R-project.org/package=usethis.

Wickham H, Danenberg P, Csárdi G, Eugster M (2021). *roxygen2: In-Line Documentation for R*. R package version 7.1.2, https://CRAN.R-project.org/package=roxygen2.

Boshnakov GN (2022). "Rdpack: Update and Manipulate Rd Documentation Objects." doi:10.5281/zenodo.3925612, R package version 2.3.

| corrio | *Apply and visualize Pearson's product-moment correlation.* |

Description

Compute all components which are part of Pearson's correlation coefficient and visualize the most important part of what is summarized in the correlation coefficient. This most important part is the difference between the values of each variable from their respective mean. While it may appear superflous for some people to visualize this part, other people may benefit from it. See vignette of this 'correlatio' package for further explanations.

Usage

```
corrio(data = NULL, visualize = TRUE)
```

Arguments

data	A data.frame with two columns, which shall be correlated by Pearson's product-moment method.
visualize	A single boolean value (default: TRUE), which determines whether the data shall be visualized.

Value

a list with a data.frame (name: dat), a list (name: details), and two graphs as elements (plot1 and plot2). dat contains these five columns:

1. x Values of the first variable (= x).
2. y Values of the second variable (= y).
3. x-mean(x) Difference between x and the mean of x.
4. y-mean(y) Difference between y and the mean of y.
5. covVec Product of x-mean(x) and y-mean(y).

details is a list with 12 objects, each of which contains an explanation as attribute:

1. Mean of variable 1 (variable 1 = x).
2. Mean of variable 2 (variable 2 = y).
3. Sum of all negative products (negSum): (x-mean(x)) * (y-mean(y)).
4. Sum of all positive products (posSum): (x-mean(x)) * (y-mean(y)).
5. Numerator of covariance formula: Sum of negSum and posSum.
6. Denominator of covariance formula: n - 1.
7. Covariance: numeratorCov/denominatorCov.
8. Standard deviation of variable 1 (i.e., x): R command sd().
9. Standard deviation of variable 2 (i.e., y): R command sd().

10. Product of standard deviations (prodSD) of variables 1 and 2 (i.e., x and y).

11. Correlation: Covariance/prodSD.

12. Percentages of pairwise directions of s, c, n (s = same, c = contrary, n = no)

plot1 and plot2 are two ways of visualizing the connection between the individual values and their respective mean value.

Author(s)

Marcel Miché

References

Curran-Everett D (2010). "Explorations in statistics: correlation." *Advances in physiology education*, **34**(4), 186–191.

Wickham H, Wickham H (2016). "Programming with ggplot2." *Ggplot2: elegant graphics for data analysis*, 241–253.

Examples

```
simData <- simcor(obs=100, rhos = .6)
corrio(data=simData[[1]], visualize = TRUE)
```

---

corvisualize	*Visualize the correlation coefficient geometrically.*

Description

Visualize the correlation coefficient geometrically, i.e., use the angle between the linear vector that represents the predictor and the linear vector that represents the outcome, show where the dropping of the perpendicular lands on the linear vector that represents the predictor in the two-dimensional linear space, finally read b regression weight from the simple linear regression between predictor and outcome; or read the beta regression weight, in case the predictor and outcome have been scaled (mean = zero, standard deviation = one).

Usage

```
corvisualize(data = NULL, x = "x1", y = "x2", visualize = TRUE)
```

Arguments

data	A data.frame with two columns, which shall be correlated by Pearson's product-moment method.
x	A single character, i.e., the column name of the data.frame which shall be the predictor (independent variable) in the simple linear regression.
y	A single character, i.e., the column name of the data.frame which shall be the outcome (dependent variable) in the simple linear regression.
visualize	A single boolean value (default: TRUE), which determines whether the data shall be visualized.

Details

Any textbook on linear algebra and/or analytic geometry usually contains at least one numeric example and a geometric visualization of a correlation between two continuous variables. I want to express my gratitude to Dr. Johannes Andres (who taught statistics as well as multivariate statistics to psychology students, of which I was one).

Value

a list with results (name: res), and one graph as elements (name: anglePlot). res is a list with 13 objects:

1. covMat Covariance matrix of predictor and outcome.
2. covPredMat Covariance matrix of predictor and the predicted outcome, based on the simple linear regression estimates.
3. corMat Correlation matrix of predictor and outcome.
4. spreadMat Square root of the variance of the predictor and the variance of the outcome. If the angle is greater than 90 degrees, the spread of the predictor is multiplied by minus one.
5. angle The angle between predictor and outcome: In R, compute: acos(cor(predictor,outcome))*180/pi.
6. rsquared Explained variance of the outcome variable.
7. errorVariance Difference between the variance of the outcome and the explained variance of the (predicted) outcome variable.
8. errorSpread Square root of the error variance.
9. observedSpread Square root of the variance of the outcome variable.
10. yhatSpread Square root of the difference between outcome variance and variance of the predicted outcome. If the angle is greater than 90 degrees, yhatSpread is multiplied by minus one.
11. bWeight The regression weight (slope) of the simple linear regression of the predictor and the outcome variable.
12. betaWeight Same as bWeight, if the predictor and the outcome have both been scaled (mean = 0, standard deviation = 1).
13. anglePlot Visualization of the regression weight, unless the function argument visualize has been set to FALSE.

Author(s)

Marcel Miché

References

Boyer CB (1949). "The invention of analytic geometry." *Scientific American*, **180**(1), 40–45.

Gniazdowski Z (2013). "Geometric interpretation of a correlation." *Zeszyty Naukowe Warszawskiej Wyższej Szkoły Informatyki*.

Graffelman J (2013). "Linear-angle correlation plots: new graphs for revealing correlation structure." *Journal of Computational and Graphical Statistics*, **22**(1), 92–106.

Graffelman J, De Leeuw J (2023). "Improved approximation and visualization of the correlation matrix." *The American Statistician*, **77**(4), 432–442.

Examples

```
positiveCorDat <- data.frame(x1=c(5,9,3,6,2,9,3,7,2,8),
                             x2=c(2,6,7,8,3,5,5,8,3,9))
negativeCorDat <- data.frame(x1=c(5,9,3,6,2,9,3,7,2,8),
                             x2=c(5,7,9,2,5,5,8,1,9,8))
# Run corvisualize with positiveCorDat.
corvisualize(data=positiveCorDat, x="x1", y="x2", visualize=TRUE)
# Run corvisualize with negativeCorDat.
corvisualize(data=negativeCorDat, x="x1", y="x2", visualize=TRUE)
```

---

lineartransform	*Linearly transform one scale into another scale.*

Description

Transform the values of a variable into other values, by using the linear model. Additionally, select the number of decimal digits of the transformed values.

Usage

```
lineartransform(futureRange = c(1, 5), vec = NULL, digits = NULL)
```

Arguments

futureRange	Vector that shows the range of the new scale, e.g., c(1, 5).
vec	A vector which contains the values that shall be transformed to the new scale.
digits	A single integer that shows the number of digits, which the transformed values shall get rounded to.

Value

a vector with the linearly transformed new values, rounded to how many digits the user has set the function argument 'digits'.

Author(s)

Marcel Miché

References

lm; linear model command from the stats package

Examples

```
someValues <- stats::rnorm(n=10)
# Linearly transform to values between 1 and 5, rounded to zero digits.
lineartransform(futureRange = c(1, 5), vec = someValues, digits = 0)
```

simcor	*Simulate two correlated variables.*

Description

Simulate pairs of variables with a predefined correlation between them.

Usage

```
simcor(obs = 100, rhos = c(-0.5, 0.5))
```

Arguments

obs
: A single integer that determines the number of simulated observations in each of the pair of variables.

rhos
: A vector with at least one value that shows the theoretical correlation between the simulated pair of variables.

Value

a list with as many data.frames (each consisting of two columns) as there are values passed to the function argument 'rhos'.

Author(s)

Marcel Miché

References

pdf, see headline: Simulating data with known correlations

Examples

```
# Simulate a list with two data.frames. The first one contains variables that are correlated
# around -.8, the second one around .7. Both data.frames contain 200 observations.
simcor(obs = 200, rhos = c(-.8, .7))
```

Index