PharmaSUG 2022 - Paper DV-056

Rita: Automated Transformations, Normality Testing, and Reporting

Daniel Mattei, TechData Service Company LLC

ABSTRACT

R is an open-source programming language that allows for highly customizable analyses and visualization capabilities within a more traditional computing environment. Statistical programmers within the clinical trials industry have taken advantage of its flexibility to create specialized packages for the creation of SDTM and ADaM data structures, tables, figures, and listings (TFLs), as well as import/export capabilities to interface with SAS datasets and associated metadata. More general packages aimed at users transitioning from SAS, providing critical functions such as descriptive statistical reports, hypothesis-testing to assess parametric assumptions of normality, and nonlinear transformations are not yet widely available for use on tabular data, as is typically the case when working with clinical data structures. Rita, a software package providing these functions, uses feature-detection algorithms to select for numeric columns (converting the type, if necessary), conduct normality testing on each field, and perform all available transformations on each column, selecting for the best-performing transformation. Rita is presented here to facilitate adoption of the package for transitioning users. This paper provides a tutorial for the features presented above, as well as Rita's several quality-of-life features, such as plotting capabilities that automatically select the most fitting way to visualize each column, detection and removal of null records, and the ability to customize which plots, normality tests, and transformations are applied if one wishes to designate these settings. Lastly, a Shiny app is provided to demonstrate Rita's features with data from the CDISC Pilot 01 Study and used to explain core functions for interested readers.

INTRODUCTION

Have you heard chatter regarding the use of R at your organization? If not, it may be upon you soon – The R programming language (R Core Team, 2021) seems to be gaining steam within the pharmaceutical industry. Many acquainted with SAS® software have begun making this transition to complement their coding of clinical data structures and tables, figures, and listings (TFLs). As budding R coders, they may be hard-pressed, however, to find commands similar to what are offered in the SAS Studio environment. Even frequent coders may find it difficult to find similar solutions.

Similarly, newcomers and veterans alike may find it tedious to write R scripts encompassing the steps involved in the process of exploratory data analysis (EDA). These scripts are not only time-consuming to write, but also often in need of large amounts of "google time" to determine the relative merits of competing methods a programmer may consider to accomplish their work, such as for visualizations, normality tests (Thode, 2002), or transformations (Osborne, 2002). This paper presents an exhaustive solution that places a dataset "one command away" from the most common EDA requirements among coders, just as a dataset might be "one PROC away" from other forms of output.

Rita is an R package for EDA available on the Comprehensive R Archive Network (https://CRAN.R-project.org/package=Rita) (Mattei and Ruscio, 2022). It is my best attempt to cure the "*Where's the PROC means?"* syndrome that may be prevalent among transitioning coders. It's also my best attempt to provide not just a foundation, but a *bridge* between the initial investigations of descriptive statistics and other aspects of EDA previously mentioned; visuals, normality testing and data transformations.

To that end, Rita uses a "one size fits all" approach prioritizing extreme ease of use that caters to novices and veterans alike. This is in the spirit of the various PROC steps employed in SAS Studio. This approach also aims to anticipate and alleviate the 'supermarket dilemma' of attempting to choose between normality tests or transformations of unclear applicability or relevance. The remainder of the introduction provides a brief outline of the package's basic process and output before diving in more deeply.

When designing the Rita() function, tabular data were the expected form of input, which is what will be focused on here (vectors are valid forms of input as well). It begins by generating descriptive statistics: the five-number summary of the minimum, maximum, quartiles, and median as well as the mean,

standard deviation, and skewness and kurtosis coefficients are presented for each variable after missing values are removed.

Next, each of five nonlinear transformations/normalizations (Osborne, 2002) are performed on each column, with the best-performing method selected and returned. The options are as follows:

(1) logarithmic transform

(2) inverse/reciprocal transform

(3) square-root transform

(4) arc-sine transform

(5) logit transform (Stevens et al., 2016)

When assessing which transformation performs best for a given column, R^2 indicating the line of best fit for corresponding pairs of transformed and theoretical normal quantiles is used from the Q-Q plot. Options are available to perform just one transformation for each column and omit this selection criterion. In addition, a sixth transformation, the Rankit (Bliss, Greenwood, and White, 1956) is provided yet omitted from this process, as it disguises between-group differences despite achieving superior normality to alternatives in a majority of cases.

Then, one of six normality tests (Shapiro-Wilk by default) (Royston 1982; Royston 1995) are performed on each **transformed** column and presented to the user. Options are available to perform alternative tests for each variable. In total the following options are available:

(1) the Shapiro-Wilk

(2) the Kolmogorov-Smirnov-Lilliefors (Lilliefors, 1967; Molin and Abdi, 1998)

(3) the Anderson-Darling (D'agostino and Stephens, 1986)

(4) D'agostino Pearson Omnibus (D'agostino and Stephens, 1986)

(5) the Jarque-Bera (Jarque and Bera, 1980)

(6) the chi-square (Moore, 1986).

Options are additionally available to perform all six tests on each transformed variable, with a Bonferroni correction applied based on the # hypotheses and desired alpha level specified in the arguments.

Lastly, a feature detection algorithm is used to visualize each raw, untransformed variable as either a histogram or a density plot. This algorithm is then used to further generate either a violin or a strip-plot for each raw variable. Density plots for each transformed column are then presented to the user, with results able to be panned within RStudio with the 'Plots' panel. Options are available to disable this automatic plotting feature and instead generate the same plots for all variables, or no plots at all. These visualizations are powered by the lattice package (Sarkar, 2008), an intuitive extension of R's base plotting capabilities to accommodate the storage of plots to be retrieved later within the R console.

The rest of this paper will focus on practical demonstrations with the mtcars dataset, and later, with questionnaire data from the first CDISC Pilot Study (Clinical Data Interchange Standards Consortium, 2007). No further mention of the underlying calculations will be made; however, all are encouraged to see *Recommended Reading* for suggestions of references.

A SIMPLE EXAMPLE: MTCARS

After installing and loading the package, let's take a quick look at the mtcars dataset before we begin. Doing so is simple, as mtcars is a built-in dataset. Here, we print the first 10 rows within the console:

```
> library(Rita)
> mtcars[1:10,]
                   mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Duster 360        14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
Merc 240D         24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
Merc 230          22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
Merc 280          19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
>
```

Display 1. A view of the mtcars dataset

EXPLORING THE RAW DATA

Rita is going to do quite a bit for us when we call its namesake function. To demonstrate its robustness, Rita() will be called with the entire mtcars dataset as an input:

```
x <- Rita(mtcars)
```

The code above produces a descriptive statistical report of each column, results of the Shapiro-Wilk test for each, and even plots of each variable in its raw and transformed form. First, we'll take a look at the report. This is going to consist of formatted output within the R console reporting a five-number summary, the mean, standard deviation, and skewness and kurtosis coefficients:

```
Desc. Stats of Raw Variable(s):
             mpg    cyl    disp      hp   drat    wt   qsec   gear  carb
Min.      10.400  4.000  71.100  52.000  2.760 1.513 14.500  3.000 1.000
1st Qu.   15.425  4.000 120.825  96.500  3.080 2.581 16.892  3.000 2.000
Median    19.200  6.000 196.300 123.000  3.695 3.325 17.710  4.000 2.000
Mean      20.091  6.188 230.722 146.688  3.597 3.217 17.849  3.688 2.812
3rd Qu.   22.800  8.000 326.000 180.000  3.920 3.610 18.900  4.000 4.000
Max.      33.900  8.000 472.000 335.000  4.930 5.424 22.900  5.000 8.000
SD         6.027  1.786 123.939  68.563  0.535 0.978  1.787  0.738 1.615
Skewness   0.705 -0.202   0.441   0.838  0.307 0.489  0.426  0.611 1.214
Kurtosis   0.194 -1.661  -0.920   0.510 -0.263 0.661  1.139 -0.737 2.370
```

Table 1. A descriptive statistical report of mtcars (all rows)

A great deal of information has been given to us, and quickly! Let's hone in on the 'mpg' column. Across all vehicles, miles per gallon (mpg) averages about 20 miles. The median seems to agree, telling us that mpg is likely distributed in a reasonably **symmetric** manner across its center if we were to plot it. The skewness, at 0.705, is a little high in the positive direction; this gives us reason to believe that that plotting mpg would reveal a slight skew of the data to the right. In other words, some of the vehicles are outliers. They have much higher mpg scores than the typical vehicle. Kurtosis is ~0.2, indicating that the tails of this distribution will not be overly heavy.

It's a good idea to see if this pans out. Fortunately, our use of Rita() has already produced plots:

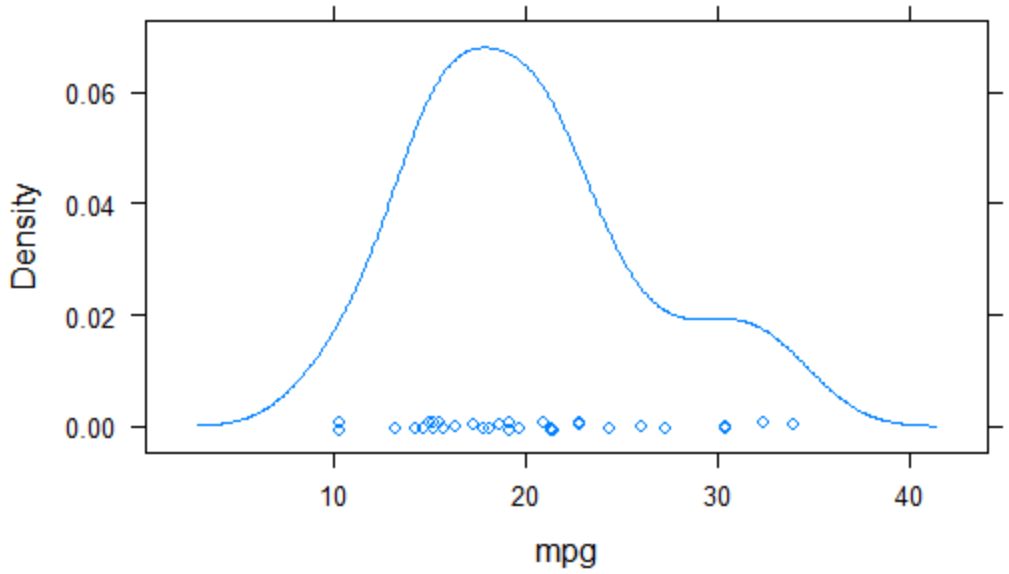


Figure 1. Density plot of raw mpg scores

For the most part, our predictions appear to be correct, with a slight hiccup. This distribution is more skewed than we gave it credit for. Moreover, we also have a better sense of the sample-size of this data, as the plot provides the datapoints as well. In addition to this density plot, a violin-plot of mpg has also been made available to us. This will provide us with visual indicators of the min + max, quartiles, and the median:

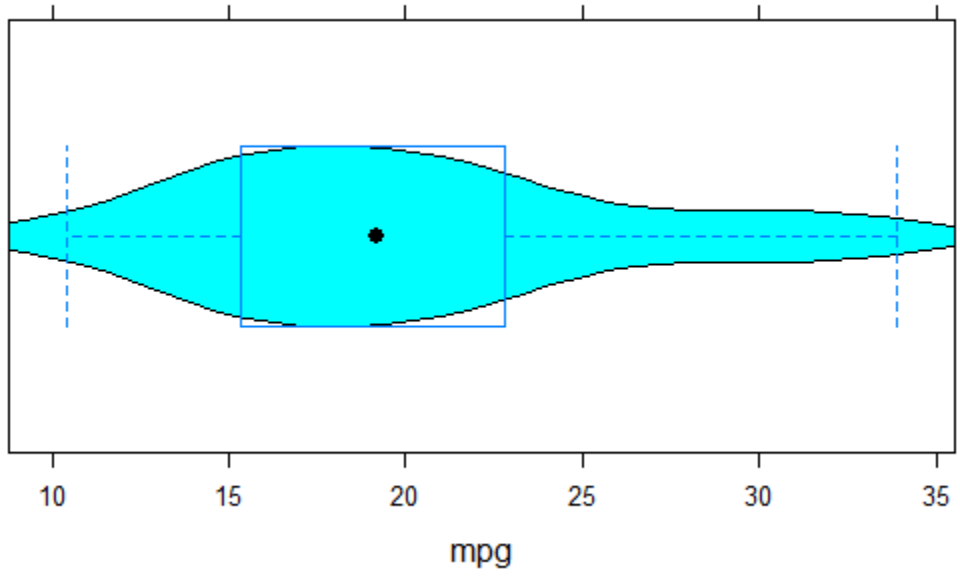


Figure 2. A violin-plot of the mpg column

With violin plots, we get the pros of box-plots (visual indicators of the min + max, quartiles, and median) as well as a good sense of the distribution of the data.

EXPLORING THE TRANSFORMED DATA

With so few records, it may be a good idea to attempt to transform the data to a more normal shape. Again, the call to Rita() has already done this for us in the previous section. Along with this plot, using Rita() has also generated density plots depicting the best-performing transformation for each variable. Let's take a look at the plot for the transformed scores of mpg:

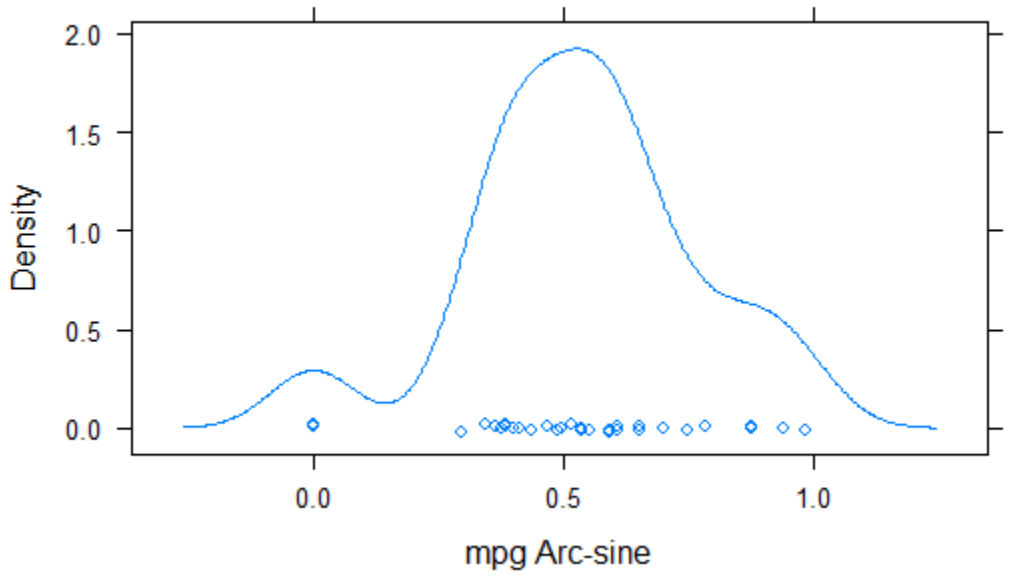


Figure 3. Density plot of the arc-sine transformed mpg scores

The arc-sine transformation is the winner! Its contenders were the log, square-root, reciprocal, and logit transformations. Because the sample-size is low (just 32 vehicles), this plot may not seem appreciably better than its raw equivalent. It's probably best to consult some supporting evidence. What are the descriptive statistics of the newly transformed scores?

Luckily, when we assigned the output of Rita() to an object (creatively named 'x' here), we received a list of all the generated plots and a dataset containing the scores of the best-performing transformation for each column. Let's access these scores and submit them to Rita():

```
y <- Rita(data = x[[1]][ ,1])
```

The aforementioned dataset was stored within the first element of the 'x' list object. With x[[1]], we are accessing this data. Then, we select for the first column of the dataset with [,1]. This works because the data are stored similarly to the raw data; just as mpg occurred first in mtcars, the transformed version of mpg, 'mpg Arc-sine', also occurs first. The data = argument is also made more explicit here. Let's see what our descriptive statistics look like:

```
Desc. Stats of Raw Variable(s):
           data
Min.      0.000
1st Qu.   0.395
Median    0.535
Mean      0.537
3rd Qu.   0.650
Max.      0.984
SD        0.227
Skewness -0.283
Kurtosis  0.997
```

Table 2. Descriptive statistics of the transformed mpg column

Skewness and kurtosis coefficients are standardized. We can compare them directly to those for the raw mpg column. We've managed to significantly improve skewness! Instead of 0.705, skewness is now much closer to 0.00, -0.283. Unfortunately, kurtosis has risen from 0.194 to 0.997. Is this enough for us to conclude that the transformation didn't improve normality by much after all? What we really need is a less subjective method of assessing the effectiveness of the transformation. It'd be a pretty good guess by now that our initial call to Rita() has already taken care of this for us. That is indeed the case:

```
Normality test results:
          mpg   cyl  disp    hp  drat    wt  qsec  gear  carb
Stat.   0.959 0.803 0.967  0.98 0.945 0.952 0.968 0.764  0.89
P-Value 0.261     0  0.42 0.795 0.106 0.163  0.45     0 0.004
Sig.        F     T     F     F     F     F     F     T     T
Test type: SW

> |
```

Display 2. Normality test results stored in 'x'

Non-significance indicates normality. According to the Shapiro-Wilk test, the newly transformed mpg column adheres to normality. One can also assess this independently of Rita(), with the SWTest() function. This is simply a wrapper for the Shapiro-Wilk test included in base R. However, other functions for all other tests mentioned in the introduction are included as well. Note that this applies as well to the provided transformations.

WRAPPING UP MTCARS

In summation, this was our first call to Rita():

```
x <- Rita(mtcars)
```

In return, we received this:

```
No. missing values omitted from each column:
mpg: 0
cyl: 0
disp: 0
hp: 0
drat: 0
wt: 0
qsec: 0
gear: 0
carb: 0

No. rows omitted with missing values: 0

Desc. Stats of Raw Variable(s):
            mpg    cyl    disp      hp   drat    wt   qsec   gear  carb
Min.     10.400  4.000  71.100  52.000  2.760 1.513 14.500  3.000 1.000
1st Qu.  15.425  4.000 120.825  96.500  3.080 2.581 16.892  3.000 2.000
Median   19.200  6.000 196.300 123.000  3.695 3.325 17.710  4.000 2.000
Mean     20.091  6.188 230.722 146.688  3.597 3.217 17.849  3.688 2.812
3rd Qu.  22.800  8.000 326.000 180.000  3.920 3.610 18.900  4.000 4.000
Max.     33.900  8.000 472.000 335.000  4.930 5.424 22.900  5.000 8.000
SD        6.027  1.786 123.939  68.563  0.535 0.978  1.787  0.738 1.615
Skewness  0.705 -0.202   0.441   0.838  0.307 0.489  0.426  0.611 1.214
Kurtosis  0.194 -1.661  -0.920   0.510 -0.263 0.661  1.139 -0.737 2.370

Normality test results:
          mpg   cyl  disp    hp  drat    wt  qsec  gear  carb
Stat.   0.959 0.803 0.967  0.98 0.945 0.952 0.968 0.764  0.89
P-Value 0.261     0  0.42 0.795 0.106 0.163  0.45     0 0.004
Sig.        F     T     F     F     F     F     F     T     T
Test type: SW

> |
```

Display 3. The complete output received with Rita(mtcars)

And the following output within the RStudio plot viewer (as individual panels):

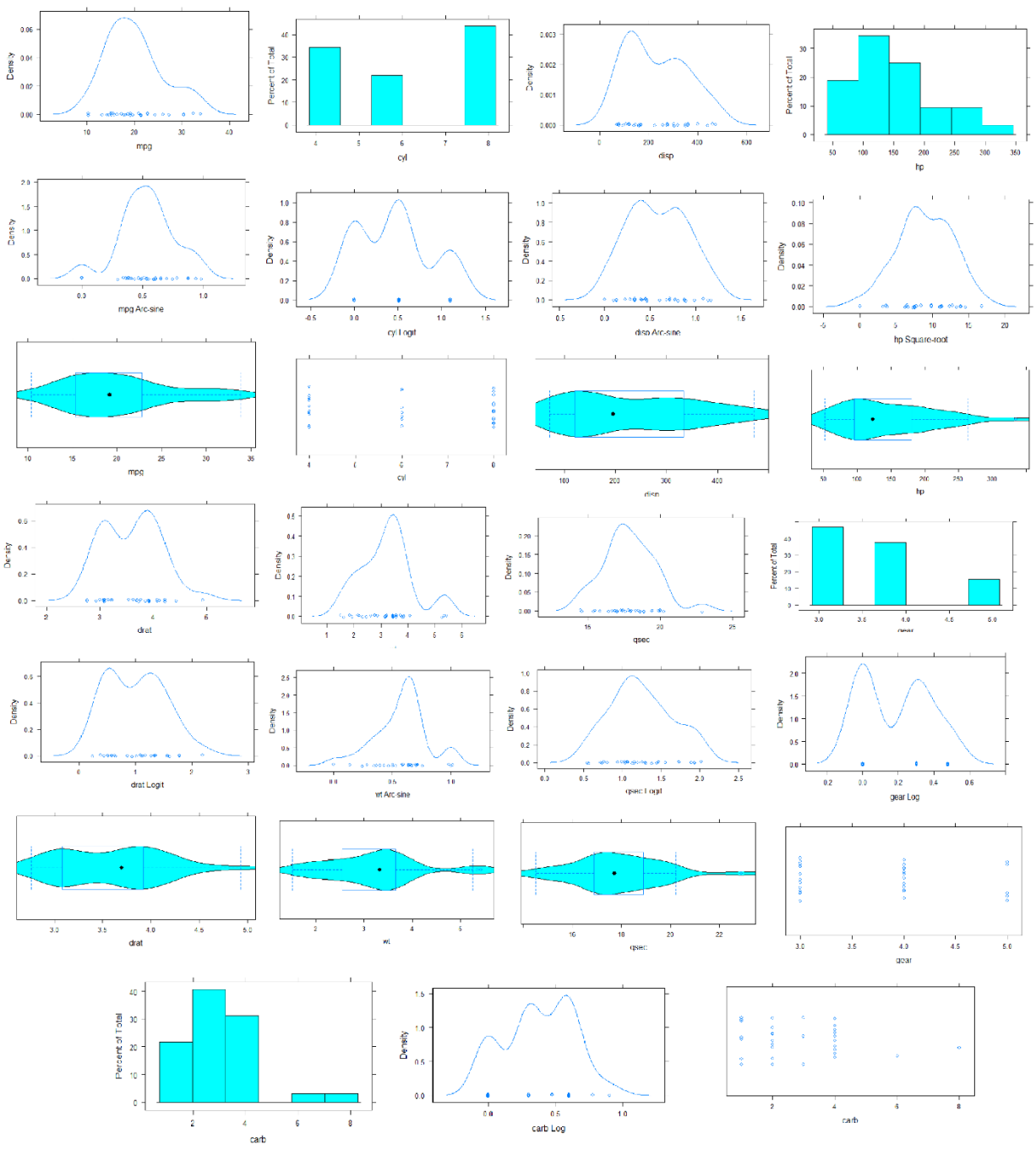
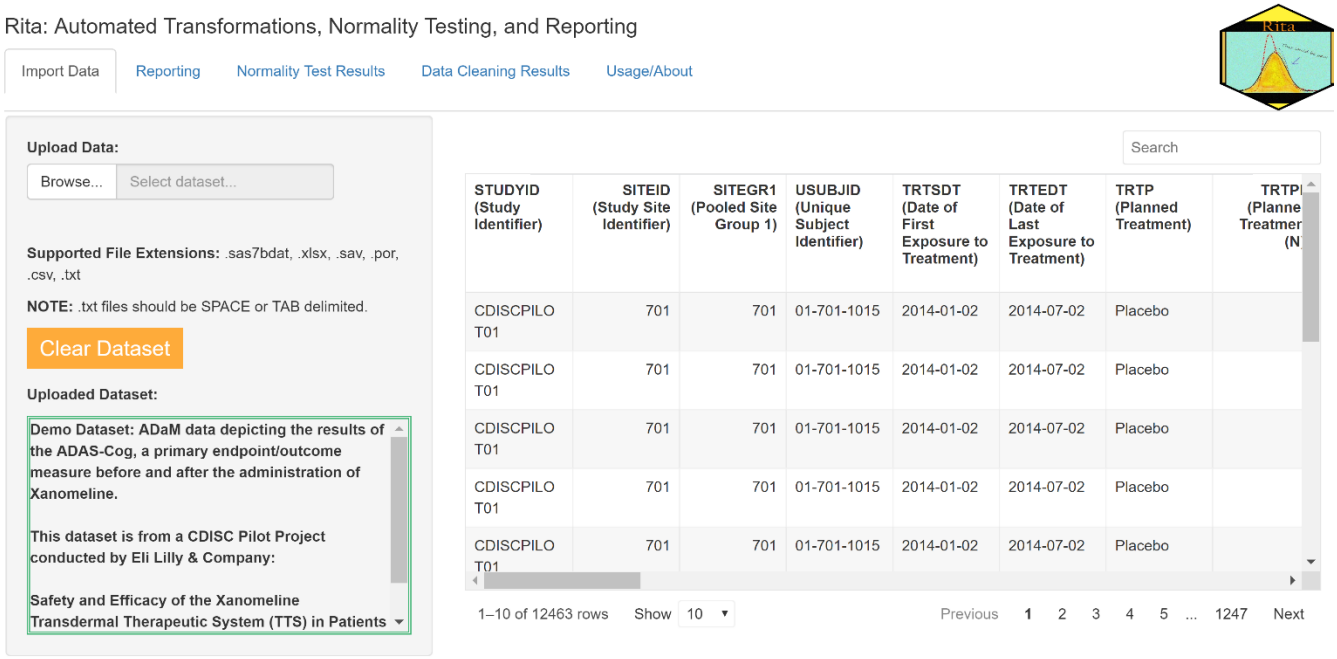


Figure 4. A composite view of all plots generated with Rita(mtcars)

Rita()'s autoPlot = argument, which equals TRUE by default, generates three plots for each column: A plot of raw scores (density or histogram), a plot of transformed scores (density), and either a violin or dot-plot. Accordingly, displayed above are three plots for each of 9 columns in the mtcars dataset. These choices are made using basic feature-detection that is detailed within the Rita package documentation (see Recommended Reading). Options are also available to generate just one type of plot, or no plots at all.

AN ADAM EXAMPLE IN SHINY

It's time to play our cards a little less close to the chest! A website is available to you, at https://danielamattei.shinyapps.io/Rita/, to allow you to test-drive Rita() for yourself without any coding required. You may upload your own non-proprietary, open-source datasets and explore it with Rita's EDA features. Initially, an ADaM questionnaire dataset, ADSADAS, is available from the CDISC Pilot Study 01 conducted by Eli Lilly and Company (CDISC, 2007) for use:



Display 4. A shiny app for Rita with the CDISC pilot study showcased

Comments or tutorials on the ADaM basic data structure (BDS) are outside the scope of this paper. In this case, scores for the Alzheimer's Disease Assessment Scale–Cognitive Subscale (ADAS-Cog) were recorded at Visit 3, with visits 8, 10, and 12 providing further datapoints. Missing post-baseline measurements were imputed with last observation carry-forward (LOCF). Change-from-baseline values, as required, were stored in the 'CHG' column.

The ADAS-Cog is a measure of cognitive function in those with Alzheimer's Disease. Xanomeline itself is an M_1 muscarinic-cholinergic receptor agonist (Bodick et al., 1997; Kueper, Speechley, and Montero-Odasso, 2018). There is an abundance of M_1 receptors within the hippocampus and cerebral cortex, to which axonal projections from the nucleus basalis of Meynart connect. These projections originating in the nucleus basalis have been observed to degenerate in Alzheimer's patients, raising the possibility that xanomeline may have a compensatory effect and preserve cognitive functioning.

WORKING WITH THE 'CHG' COLUMN

As demonstrated previously, we've seen that Rita() may work on all columns submitted to the data = argument. We'll examine the change of baseline of Adas Cog-11 subscores among those randomized to high doses of xanomeline. This step can be completed within the website, as well as the code. We will assume they have already been completed and run Rita() on the 'CHG' column:

```
x <- Rita(ADQSADAS$CHG, test = 4)
```

We've changed things up by specifying the test argument to = 4, which corresponds to the Jarque-Bera test for normality. The Jarque-Bera test prioritizes the assessment of skewness and kurtosis in determining normality. After examining the descriptive statistics of our data, this decision makes sense:

```
Desc. Stats of Raw Variable(s):
            data
Min.     -11.000
1st Qu.   -1.000
Median     1.000
Mean       1.124
3rd Qu.    3.000
Max.      13.000
SD         3.859
Skewness   0.298
Kurtosis   1.567

Normality test results:
          data
Stat.   56.025
P-Value   0.55
Sig.         F
Test type: JB

Warning message:
In JBTest(data, alpha, j) :
  N < 2000: Output p-values obtained via bootstrapping.
> 
```

Display 5. Results of Rita() for the CHG column

Excess kurtosis has exceeded 1, indicating fattier tails in the distribution. Skewness is not particularly worrying in this instance, but a drift of scores may be noted in our visual assessment of the density plot and violin plot:

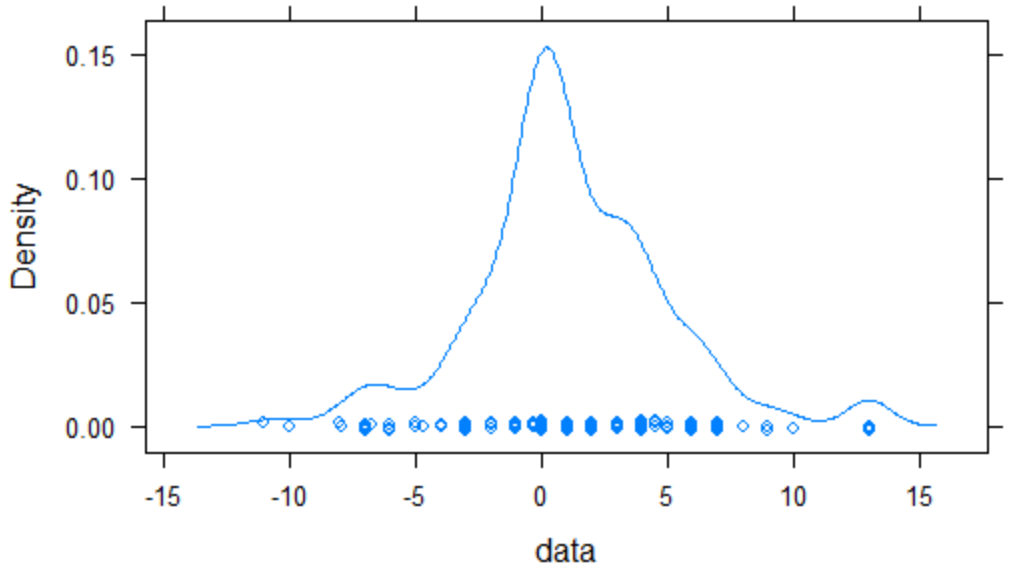


Figure 5. Distribution of raw scores for the CHG column

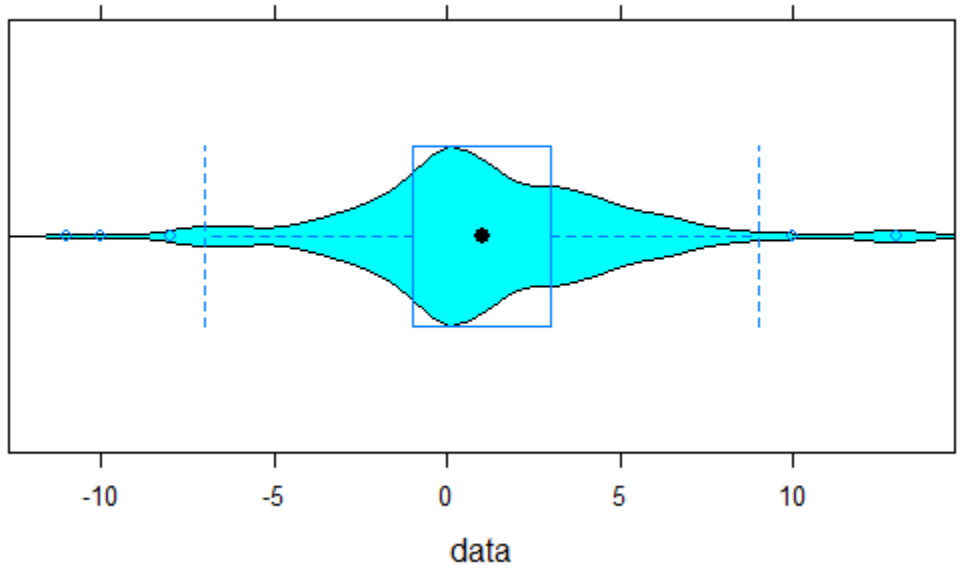


Figure 6. Violin plot of the CHG column

Luckily, as we saw prior to these visuals (Display 5), the Jarque-Bera normality test had reported that the transformed CHG column is not statistically significant. This indicates that the data are normal (per recommendations of the literature, this result was obtained empirically; see Rita documentation in Recommended Reading). Let's take a look at the plot of the transformed values:

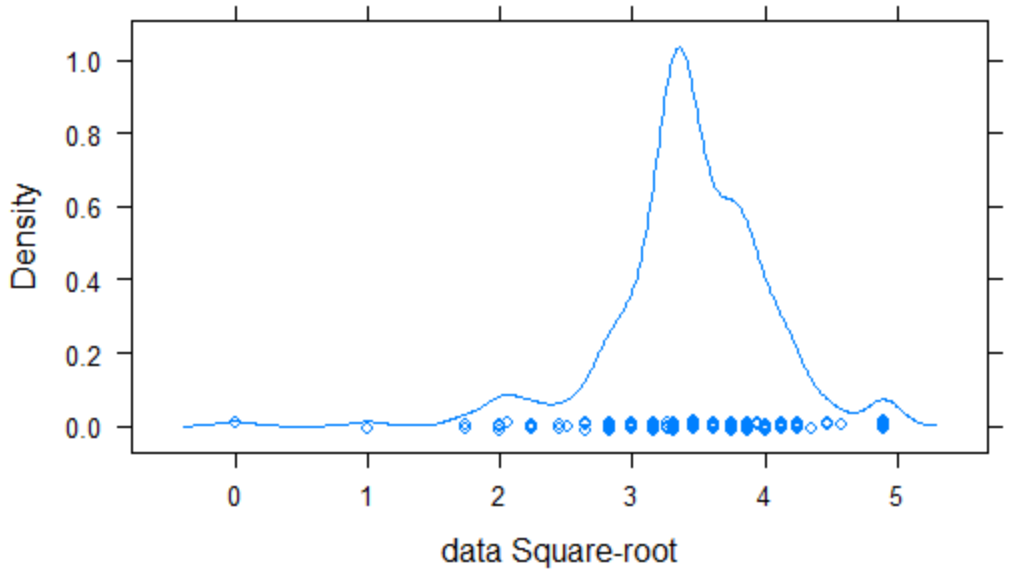


Figure 7. Distribution of the transformed values of the CHG column

Although the center of the distribution has shifted, most values have been pulled closer together and away from the tails. This is a marked improvement from the raw plot, but a few outliers do remain. If we have a theoretical justification for doing so, we may choose a different transformation with the xform = argument. Let's try things out with the Rankit transformation:

```
x <- Rita(ADQSADAS$CHG test = 4, xform = 7)
```

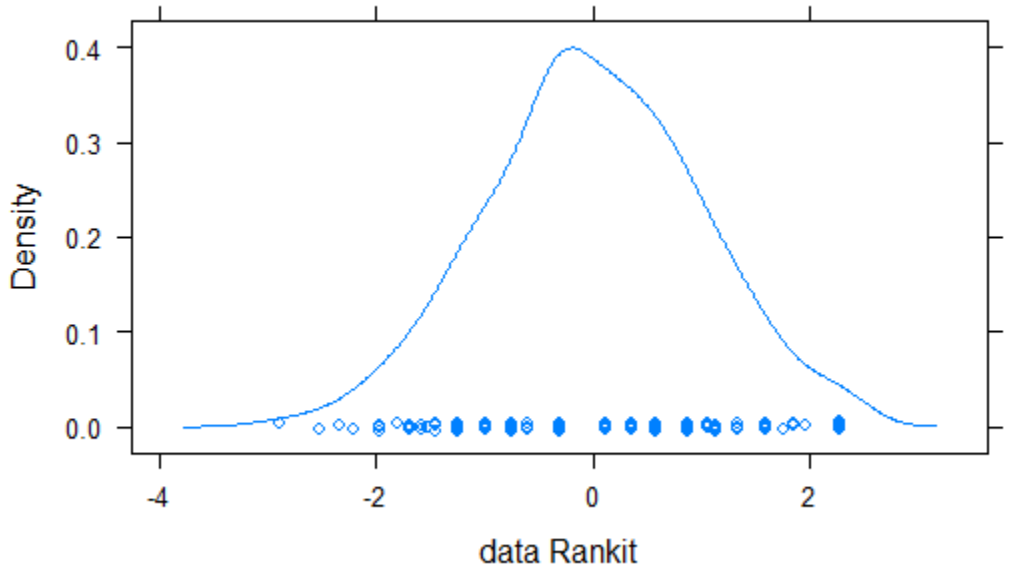


Figure 8. The effect of the Rankit transformation on the CHG column

Much better! Unfortunately, the Rankit makes inferential comparisons based on the mean as a point-estimate extremely difficult, if not impossible, when it is applied to all groups prior to analysis. It is, however, perhaps the most robust and effective non-linear transformation available when this is not an issue.

CONCLUSION

In all, the Rita package provides a suite of descriptive statistical reporting, normality-testing, and non-linear transformations to accommodate exploratory data analysis, corner to corner. The CRAN code itself is appropriate when used ad-hoc for exploratory purposes, as well as the accompanying Shiny app showcasing its use.

For industrial applications, please be aware that Rita's code is hosted on a GitHub mirror that provides service to the Comprehensive R Archive Network (CRAN), where Rita is available. Although advisable to install the package within an IDE, the code for Rita may be inspected here: https://github.com/DanielAMattei/Rita, which may be warranted for validation purposes. It is again recommended to visit Rita's CRAN page here: https://CRAN.R-project.org/package=Rita, for formal metadata and to access its documentation, which further specifies the algorithms used and directs the user to appropriate sources (see also the reference section and Recommended Reading).

Using the website is never appropriate for any business purpose; but the code is open-source and freely available, as mentioned, under an MIT license.

REFERENCES

Bodick, N. C., Offen, W. W., Shannon, H. E., Satterwhite, J., Lucas, R., van Lier, R., & Paul, S. M. (1997). The selective muscarinic agonist xanomeline improves both the cognitive deficits and behavioral symptoms of Alzheimer disease. *Alzheimer disease and associated disorders, 11 Suppl 4,* S16–S22.

Bliss, C. I., Greenwood, M. L., & White, E. S. (1956). A rankit analysis of paired comparisons for measuring the effect of sprays on flavor. *Biometrics, 12*(4), 381-403.

Clinical Data Interchange Standards Consortium (2007). CDISC SDTM/ADaM pilot project. https://bitbucket.cdisc.org/projects/CED/repos/sdtm-adam-pilot-project/browse

D'agostino, R. B., & Belanger, A. (1990). A Suggestion for Using Powerful and Informative Tests of Normality. *The American Statistician, 44(*4), 316–321. https://doi.org/10.2307/2684359

D'agostino, R. B., & Stephens, M. A. (1986). Goodness-of-fit-techniques (Vol. 68). CRC press.

Jarque, C. M. and Bera, A. K. (1980). Efficient test for normality, homoscedasticity and serial independence of residuals*. Economic Letters, 6*(3), pp. 255-259.

Kueper, J. K., Speechley, M., & Montero-Odasso, M. (2018). The Alzheimer's Disease Assessment Scale-Cognitive Subscale (ADAS-Cog): Modifications and Responsiveness in Pre-Dementia Populations. A Narrative Review. *Journal of Alzheimer's disease : JAD, 63*(2), 423–444. https://doi.org/10.3233/JAD-170991

Lilliefors, H.W. (1967). On the Kolmogorov-Smirnov Test for Normality with Mean and Variance Unknown. *Journal of the American Statistical Association, 62,* 399-402.

Mattei, D., & Ruscio, J. (2022). Rita: Automated Transformations, Normality Testing, and Reporting. R package version 1.2.0. https://CRAN.R-project.org/package=Rita

Molin, P., & Abdi, H. (1998). New Tables and numerical approximation for the KolmogorovSmirnov/Lillierfors/Van Soest test of normality.

Moore, D.S., (1986) Tests of the chi-squared type. In: D'agostino, R.B. and Stephens, M.A., eds.: Goodness-of-Fit Techniques. Marcel Dekker, New York.

Osborne, J. W. (2002). Notes on the use of data transformations. *Practical Assessment, Research and Evaluation, 9(1),* 42-50.

Osborne, J. W. (2002). The Effects of Minimum Values on Data Transformations. Retrieved from https://files.eric.ed.gov/fulltext/ED463313.pdf

Patrick Royston (1982). Algorithm AS 181: The W test for Normality. *Applied Statistics, 31,* 176–180. 10.2307/2347986

Patrick Royston (1982). An extension of Shapiro and Wilk's W test for normality to large samples. *Applied Statistics, 31,* 115–124. 10.2307/2347973

Patrick Royston (1995). Remark AS R94: A remark on Algorithm AS 181: The W test for normality. *Applied Statistics, 44,* 547–551. 10.2307/2986146

Peng, B., Robert, K. Y., DeHoff, K. L., & Amos, C. I. (2007, December). Normalizing a large number of quantitative traits using empirical normal quantile transformation. In BMC proceedings (Vol. 1, No. 1, p. S156). BioMed Central. doi: 10.1186/1753-6561-1-s1-s156

R Core Team (2021). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/

Sarkar, Deepayan (2008) Lattice: Multivariate Data Visualization with R. Springer, New York. ISBN 978-0-387-75968-5

Shreve, Joni N. and Donna Dea Holland. 2018. SAS® Certification Prep Guide: Statistical Business Analysis Using SAS®9. Cary, NC: SAS Institute Inc.

Soloman, S. R., & Sawilowsky, S. S. (2009). Impact of rank-based normalizing transformations on the accuracy of test scores. *Journal of Modern Applied Statistical Methods, 8*(2), 9.

Stevens, S., Valderas, J. M., Doran, T., Perera, R., & Kontopantelis, E. (2016). Analysing indicators of performance, satisfaction, or safety using empirical logit transformation. *BMJ (Clinical research ed.), 352,* i1114. https://doi.org/10.1136/bmj.i1114

Thode, H.C. (2002). Testing For Normality (1st ed.). CRC Press. https://doi.org/10.1201/9780203910894

Warton, D. I., & Hui, F. K. (2011). The arcsine is asinine: the analysis of proportions in ecology. *Ecology, 92(*1), 3-10.

ACKNOWLEDGMENTS

The author would like to thank and acknowledge the late John W. Tukey, who first coined the phrase "exploratory data analysis." "Rita" is the phonetic pronunciation of "REDA," or R Exploratory Data Analysis, and relies on the spirit of Dr. Tukey's approach now widely treasured in the statistical sciences.

RECOMMENDED READING

- *Package 'Rita' Documentation*
- *Testing for Normality, 1st Ed. (Thode)*
- *Notes on the Use of Data Transformations (Osborne)*

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Name: Daniel Mattei
E-mail: DMattei@live.com
Website: https://danielamattei.shinyapps.io/Rita/