Essentials of Statistics for the TI-Nspire CX CAS Handheld and Associated Emulators.

Language: English

Version: 1.0

Date: 3/4/2020

Author: James O. Thompson

Email: jodaddy101@hotmail.com

Table of Contents

1. Introduction

The functions presented here are not intended to teach statistics theory. Instead they should complement a good text on the subject. Such a text can be found in “Essentials of Statistics” by Mario F. Triola. This book is frequently used in a college course for students majoring in business, management, economics, etc.

Any student attempting to learn statistics without, at least, help with a modern calculator such as the TI-84 is sorely disadvantaged. Even with the help of the TI-84, the flood of trivial calculations obscures the beauty of statistics. Burdened with repetitive, redundant obscuring calculations such a disadvantaged student would be able to solve few of the many rich exercises at the end of each section of the book. With the help of the functions presented here the student should be able to solve many of the interesting problems posed at the end of each book section. That … and have time for other concurrent college courses as well.

The functions presented here are for the student equipped with a TI-Nspire CX CAS handheld or an emulator on a PC, iPad, etc. The “CAS” (Computer Algebra System) part is essential. The functions cannot be installed on the TI-Nspire without the CAS part.

The functions should be introduced piece meal through the course as aids to automate methods already taught from the book. The student should know the process being automated and use the automation only to avoid repetitive calculations not contributing to the learning process.

2. Platforms

The functions presented here will execute on the TI-Nspire CX CAS handheld, the TI-Nspire CX CAS Student Software and associated emulators executing on the iPad.

3. Installation

All of these functions are bundled into a single file, “stats.tns”. This file is to be installed in the “mylib” folder where it will join the other files (numtheory.tns and linalgcas.tns) from the default installation. The file can be transferred to the handheld using the process described for file transfers in the handheld owner’s manual. On a PC the “mylib” subdirectory is in the User’s Documents subdirectory which, unfortunately, Microsoft likes to hide. See this web reference on how to locate the User’s Documents subdirectory.

After installing this file, you must “Refresh Libraries”. On the handheld, press Doc, then choose the “Refresh Libraries” option. On the Student Software, choose the Tools menu, then the “Refresh Libraries” option. Following this the functions may be accessed as the file

name (stats) followed by the backslash (\) followed by the function name. e.g., “stats\about()”.

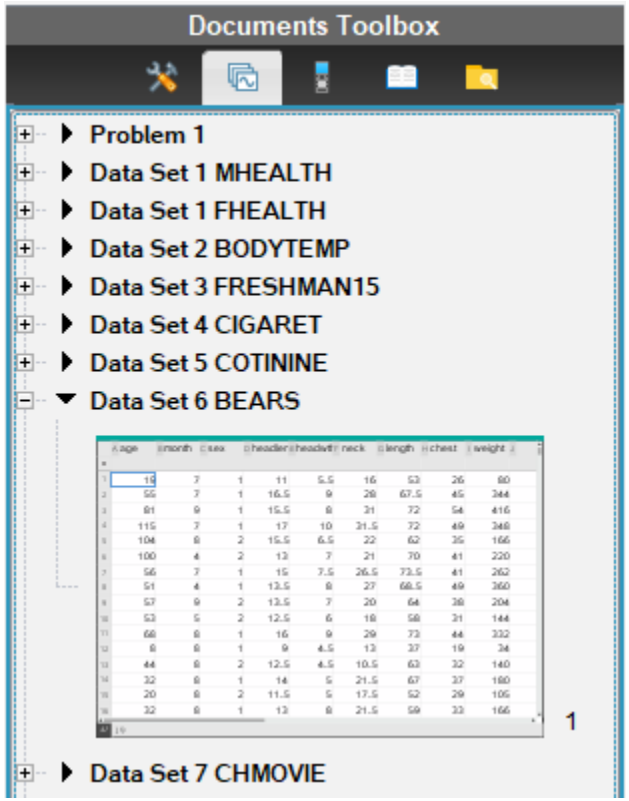
Following installation, the function source code may be inspected and modified as desired. Furthermore, it may be distributed freely and for free.

4. Data Sets

The “Essentials of Statistics” textbook by Triola includes a rich collection of data sets for many platforms such as Excel, Minitab, etc. But not for the TI-Nspire. All of those data sets are contained in the stats.tns file.

Each data set (e.g., Data Set 6 Bears) has been imported into a spreadsheet application in a dedicated TI-Nspire problem named for the data set. Each column of the spreadsheet is one of the data lists in the dataset. If you open the Page Sorter in the Documents Toolbox in stats.tns you will see a list of the problems. You can copy any of the spreadsheets, then paste it into your problem as a spreadsheet application. Each column will automatically become a list in your problem. You may then delete the spreadsheet and the lists will persist.

a. Example illustrating the Page Sorter of the Documents Toolbox stats.tns problem for Data Set 6 BEARS.



5. Function Listing in Book Order

These functions are best introduced in the same order as the book. Here is a matrix listing the functions in that order. The functions not specific to a section are helper functions used by the other functions.

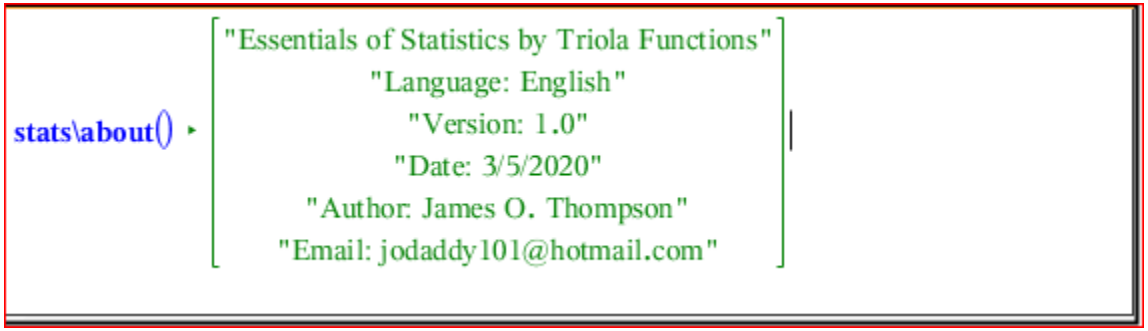
Function	See Book	Section Title
percentextrap	Section 03-4	Measures of Relative Standing and BoxPlots.
percentiletoval	Section 03-4	Measures of Relative Standing and BoxPlots.
valtopercentile	Section 03-4	Measures of Relative Standing and BoxPlots.
bayes	Section 04-7	Bayes Theorem
probdist	Section 05-2	Random Variables
sampmeanscdf	Section 06-5	The Central Limit Theorem
sampmeanspdf	Section 06-5	The Central Limit Theorem
sizepopprop	Section 07-2	Estimating a Population Proportion
critical_chi2	Section 07-5	Estimating a Population Variance
critical_z	Section 07-5	Estimating a Population Variance
interval_chi2	Section 07-5	Estimating a Population Variance
sizepopmean	Section 07-5	Estimating a Population Variance
beta_prop	Section 08-2	Basics of Hypothesis Testing
power_t	Section 08-2	Basics of Hypothesis Testing
test_chi2	Section 08-6	Testing a Claim About a Standard Deviation or Variance
critical_r	Section 10-2	Correlation
anyties	Section 10-5	Rank Correlation
critical_rs	Section 10-5	Rank Correlation
rank_cor	Section 10-5	Rank Correlation
rankize	Section 10-5	Rank Correlation
about		
getmsg		
matinsert		
sortlist		
sortmat		
trim		

6. Function Definitions in Alphabetical Order

This section contains a definition of all functions in alphabetical order.

a. about()

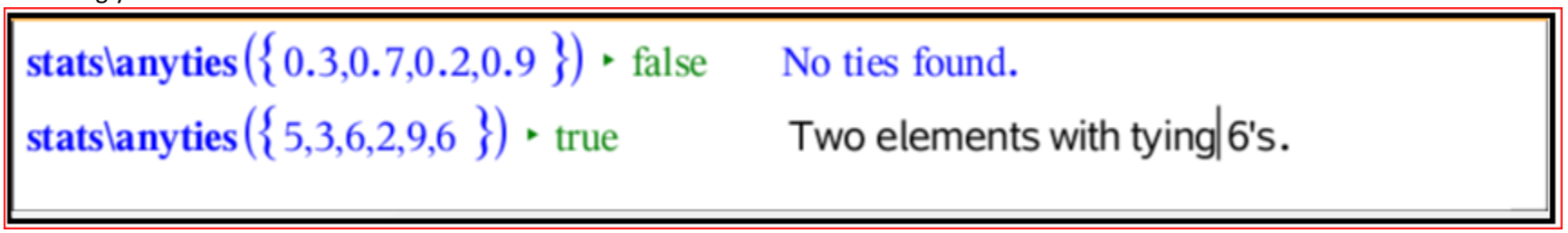
The function gives some development data for the library.



b. anyties(list)

The algorithm for Rank Correlation differs if some of the data has ties. See book Section 10-5 for the discussion.

This function will accept data in a list, sort the data, then index through the list looking for tying entries. It returns True or False accordingly.

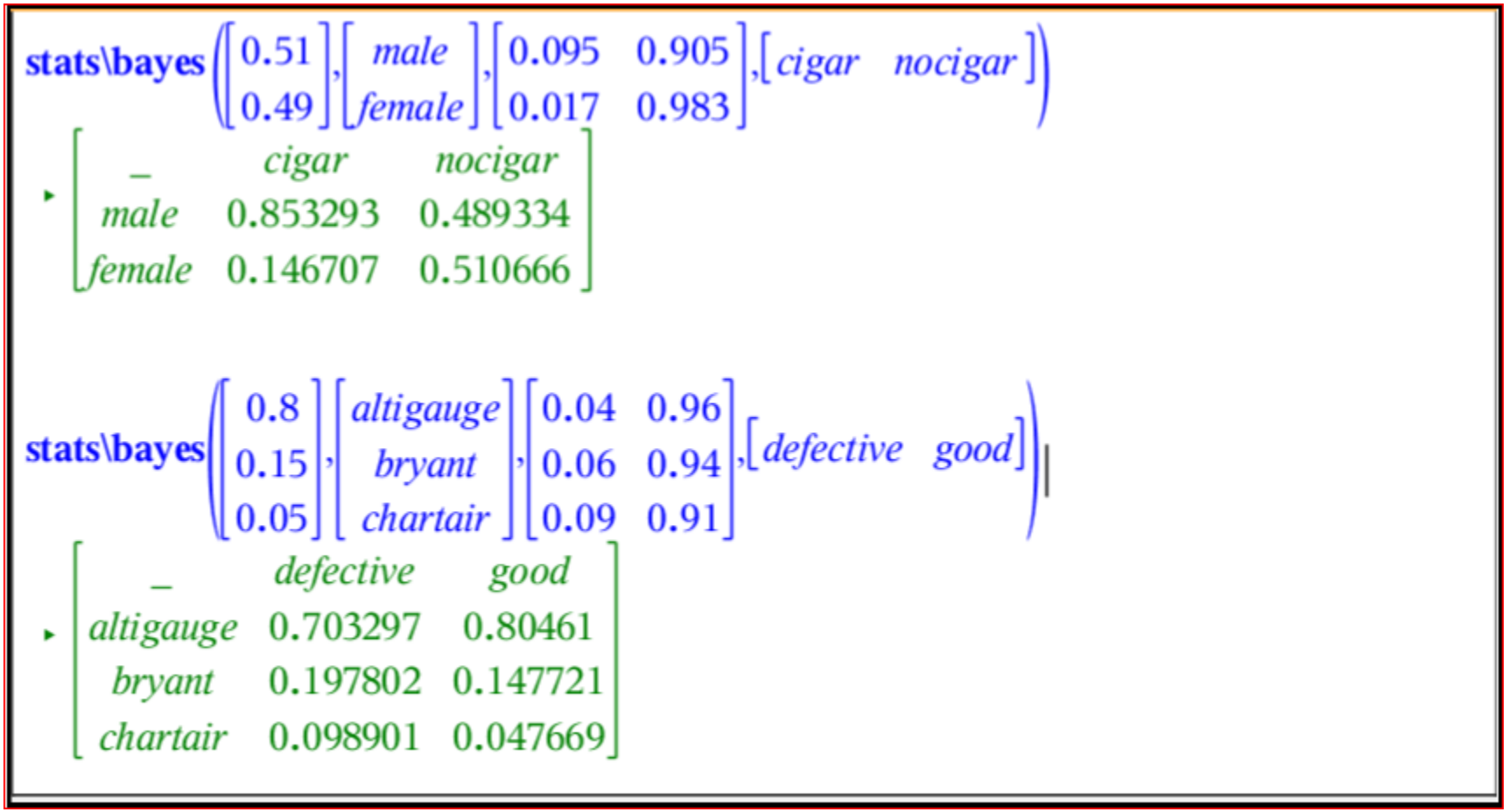


c. bayes(ev1,ev1labels,ev2,ev2labels)

See book Section 4-7 and CD-ROM for Bayes' Theorem.

The inputs to this function consists of 4 matrices. The second and fourth matrices are labels for the variable probabilities. The first matrix is a column matrix where each row element is the probability for the associated label. The third matrix is a set of conditional probabilities where the rows are associated with the rows of the first matrix and the columns are a different set of conditional probabilities.

The inputs and outputs are best understood by the following examples from the CD-ROM implementing Bayes Theorem.



d. beta_prop(p1,p2,n,α)

Compute β (Type II error) for a proportion test. This function is used in Section 8-2, exercise 47. See pg. 399 for graph of beta and power.

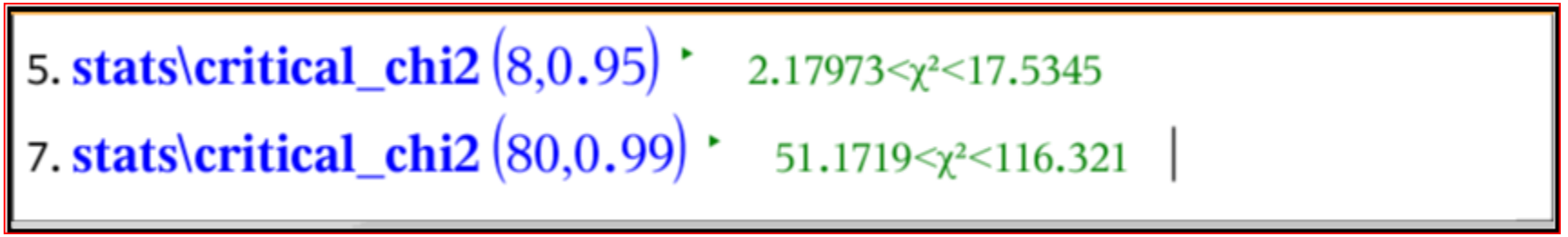
Arguments: p1 is the population proportion, p2 is the statistic population, n is the number of samples and α is the significance level.

For example:

e. critical_chi2(df,cl)

Function to compute the critical points in a chi2 distribution. See book Section 7-5 for this function. The arguments are: degrees of freedom and confidence level. Note different values for the left and right tail because the chi2 distribution is not symmetrical.

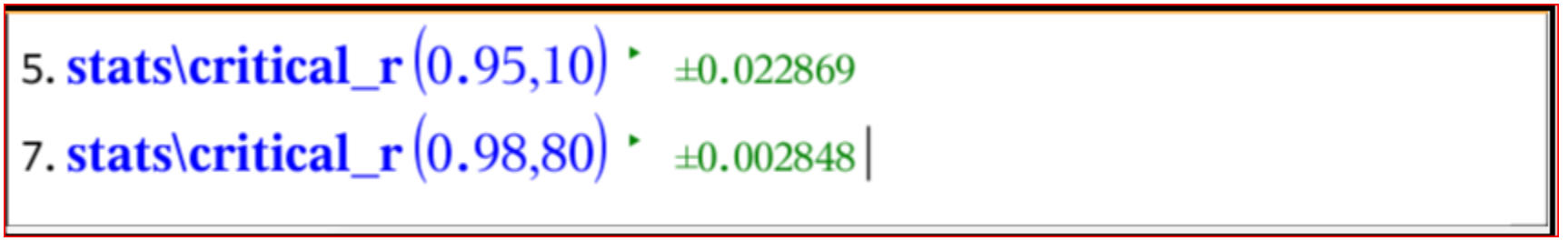
For example:



f. critical_r(α,n)

This function will compute critical values for the Pearson correlation coefficient R. It replaces Table A-5 first used in Chapter 10-2. This equation comes from solving the t(r) equation in Chapter 10-2 for r.

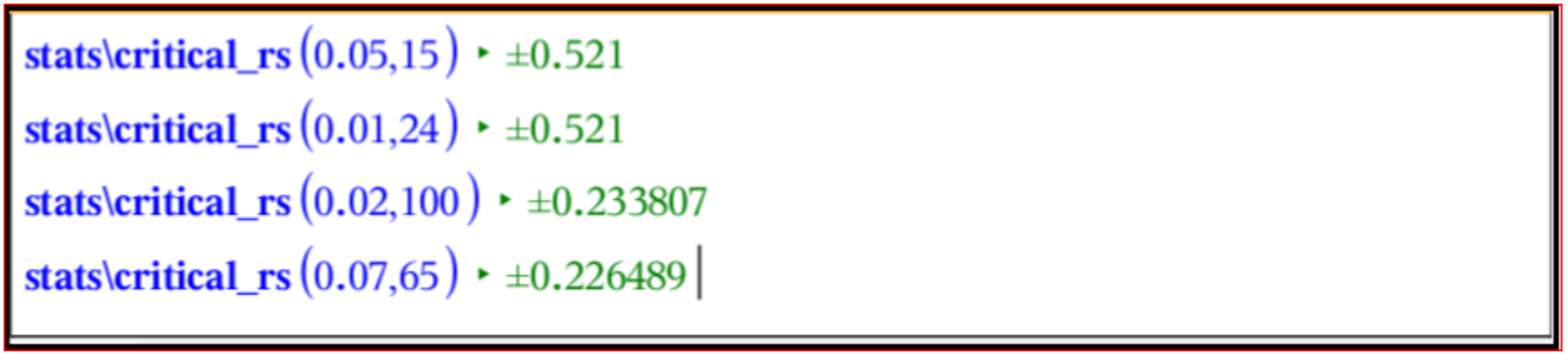
The arguments: α is the significance level and n is the number of sample pairs.



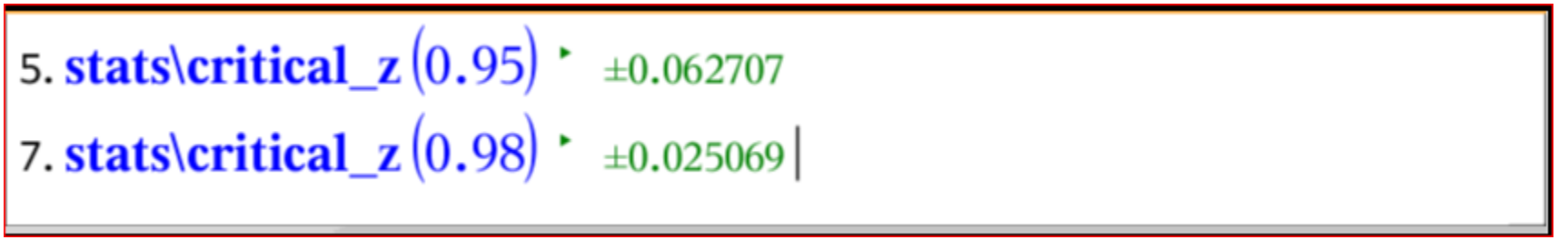
g. critical_rs(α,n)

This function will compute critical values for the Spearman's rank correlation coefficient R. It replaces Table A-6 and the general

algorithm first used in Chapter 10-5. Inputs are the significance level and number of data entries.



h. critical_z(α)
Function to compute the critical points in a normal distribution. See book Section 7-5 for this function. The argument α is the significance level.



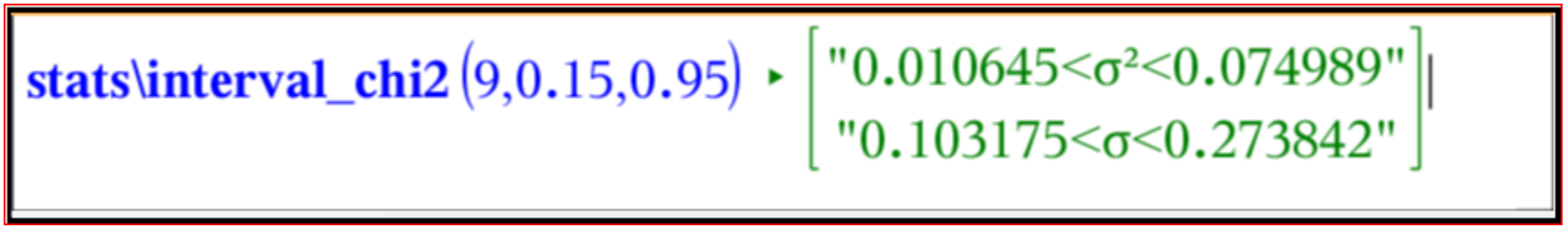
i. getmsg(index)
This function is used internally and not for public use. It generates error messages according to an index.

For example, if the user invokes the beta_prop() function with the wrong argument type he will get the following error message.

j. interval_chi2(df,s,cl)
Function to compute the confidence level for the population standard deviation and variance from a sample statistic. See book Section 7-5 for this function.

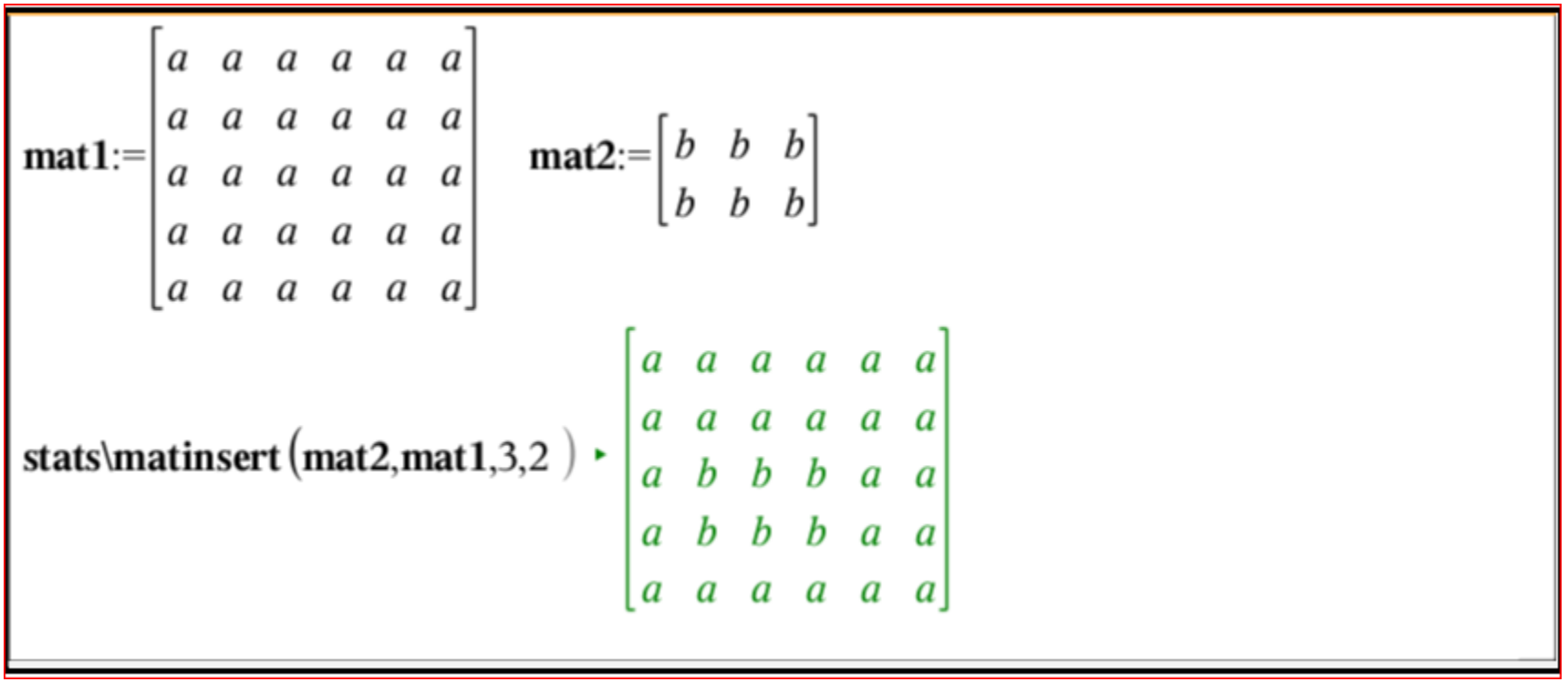
The arguments are: degrees of freedom (usual the sample size minus 1), the sample standard deviation and the confidence level.

Example:



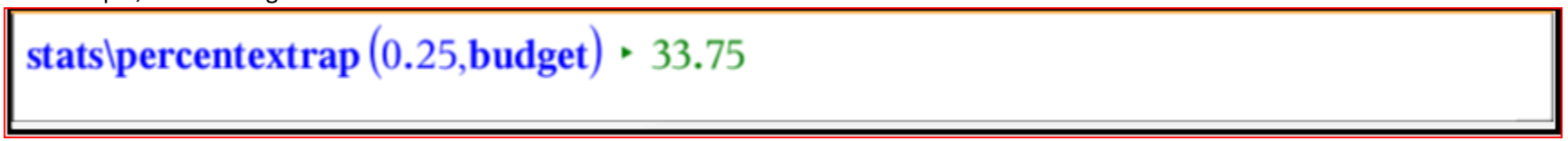
k. matinsert(mat1,mat2,row,col)
This function is used internally and not for public use. It inserts one matrix into another.

For example:



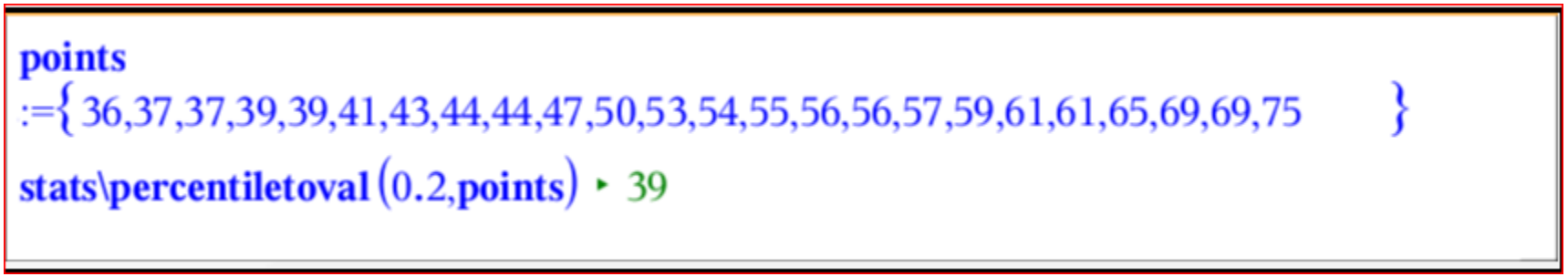
l. percentextrap(k,vallist)
The function will convert the kth percentile to the nearest extrapolated list data value. See book Section 3-4 Exercise 37 for a description of this function.

For example, where Budget is from Data Set 9 Movies:



m. percentiletoval(k,vallist)
This function will convert the kth percentile to the nearest list data value. See book Section 3-4 for definition and usage.

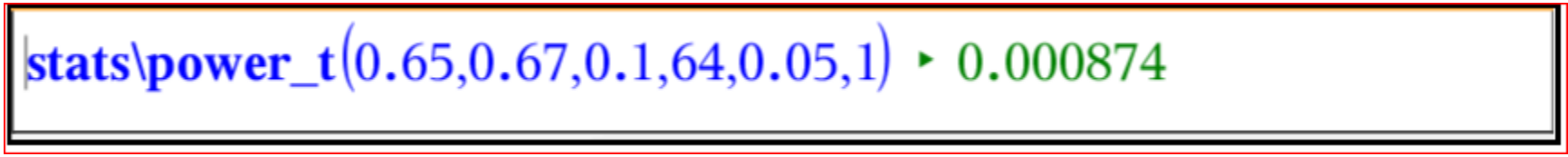
Exmple:



n. power_t(μ,x,s,n,α,hypot)
Compute the power for a t distribution test. See book Section 8-2 Exercise 47 for a graph illustrating this function.

The arguments are: μ is the population mean; x is the sample mean; s is the sample standard deviation; n is the sample size; α is the significance level; and hypot is the hypothesis tail. Hypot is negative for a left tail, zero for two tails and positive for a right tail.

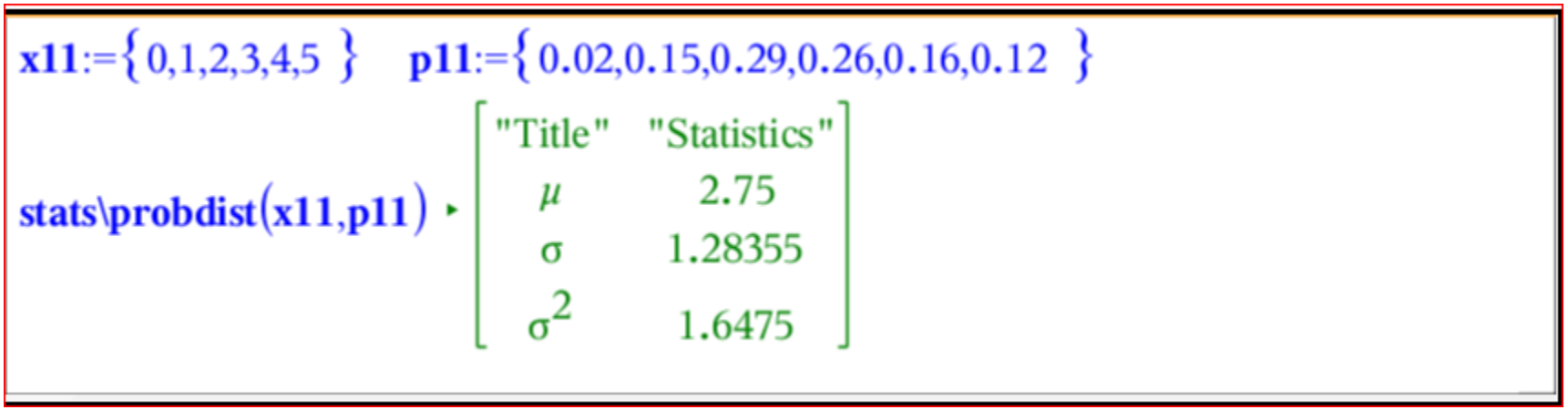
Example:



o. probdist(xlist,plist)
This function will compute some parameters for a distribution. See book Section 5-2 for this calculation.

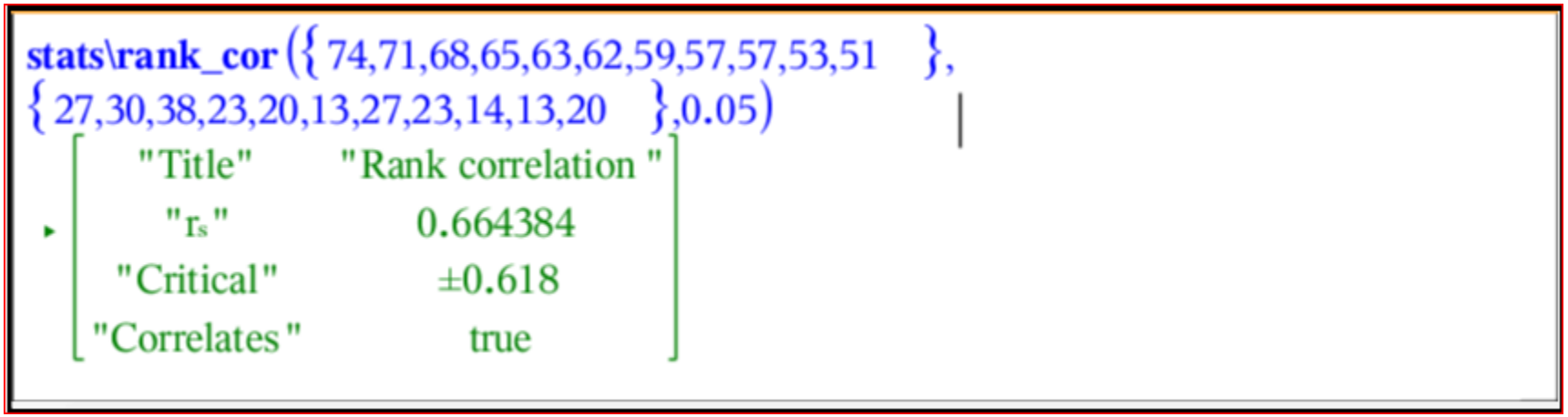
Arguments: xlist is a list of values and plist is a list of probabilities (not frequencies) for the associated values.

Example:



p. rank_cor(x,y,α)

This function will detect rank correlation. Inputs are a list of x and y datum and a significance level. See Section 10-5 for these formulas.



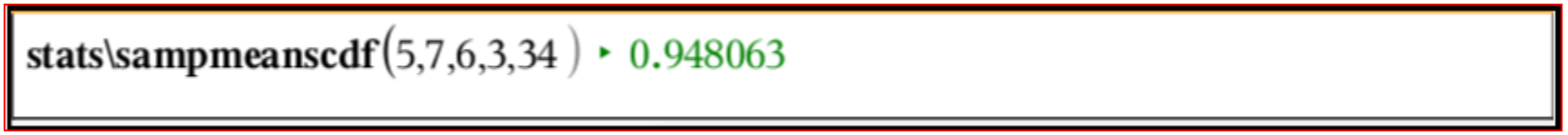
q. rankize(list)

This function will convert a list of values to sorted ranks. Tying entries are set to the mean of the tie values. The data order will

not change. See book Section 10-5 for this function.

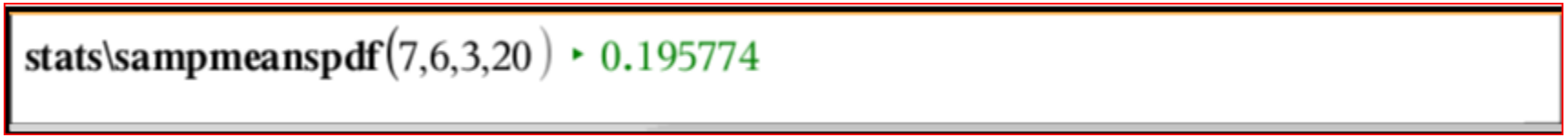
r. sampmeanscdf(lbound,ubound, μ,σ,n)
Compute the cumulative probability for sample means of size n. See book Section 6-5 for this formula.

Example:



s. sampmeanspdf(x,μ,σ,n)
Compute the probability for sample means of size n. See book Section 6-5 for this formula.

Example:



t. sizepopmean(clevel,σ,me)
Compute the sample size for a population mean estimate. See book Section 7-3 for this formula.

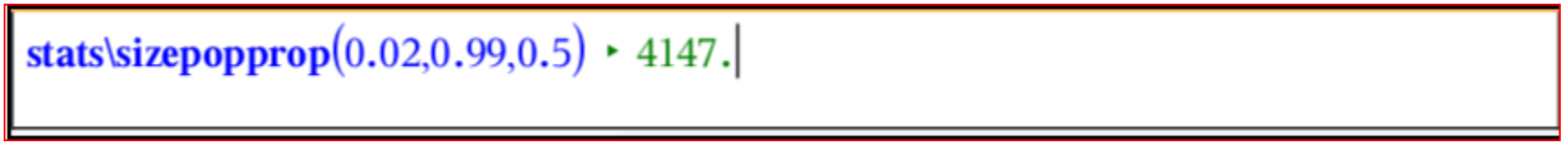
The arguments are: clevel is confidence level; σ is the standard deviation; and me is the margin of error.

Example:

u. sizepopprop(me,clevel,p)
Compute the sample size to estimate a population proportion. See book Section 7-2 for this formula.

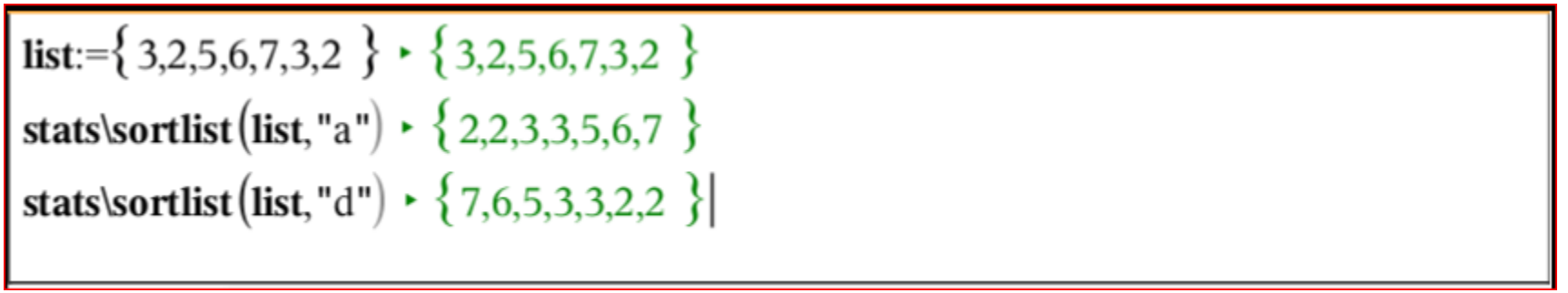
The arguments are: me is margin of error; clevel is the confidence level; and p is the sample proportion.

Example:



v. sortlist(listin,order)
This function is used internally and not for public use. It sorts a list into ascending or descending order. The order argument is a string of either “a” for ascending or “d” for descending.

For example:



w. sortmat(mat,col)
This function is used internally and not for public use. It sorts the matrix rows by any column of the matrix.

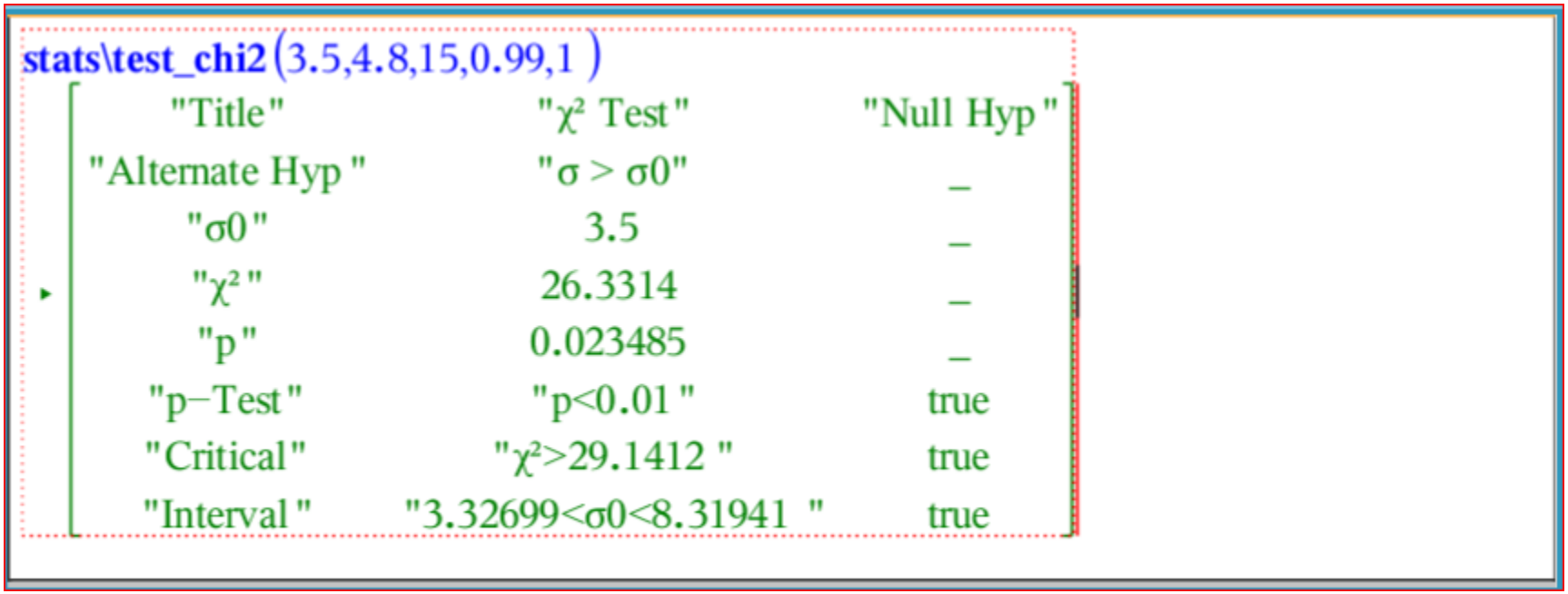
For example:

$$\text{matrix} := \begin{bmatrix} 1 & 3 & 2 \\ 5 & 6 & 1 \\ 9 & 8 & 3 \\ 1 & 4 & 5 \end{bmatrix} \quad \text{stats\sortmat}(\text{matrix}, 2) \rightarrow \begin{bmatrix} 1 & 3 & 2 \\ 1 & 4 & 5 \\ 5 & 6 & 1 \\ 9 & 8 & 3 \end{bmatrix}$$

x. test_chi2(σ0,s,n,cl,hypot)

This function implements Section 8-6 testing claim on σ or σ^2. The arguments: σ0 is the claimed population variance; s is the sample standard deviation; n is the sample size; cl is the confidence level; and hypot is the hypothesis. Hypot is negative for a left tailed test, zero for a two tailed test, and positive for a right tailed test.

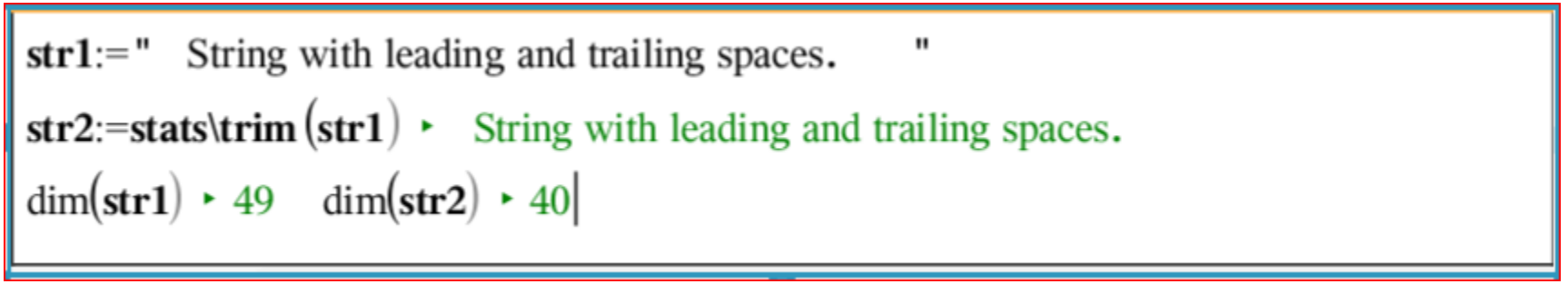
Example:



y. trim(str)

This function is used internally and not for public use. It removes leading and trailing spaces from a string.

For example:



z. valtopercentile(val,vallist)

This function will compute the percentage of list items less than val. See book Section 3-4 for its description.

Example:

points
$:=\{36,37,37,39,39,41,43,44,44,47,50,53,54,55,56,56,57,59,61,61,65,69,69,75\}$

stats\valtopercentile$(47,$**points**$)$ ▸ 0.375