Name:

2 pm to 5:50 pm March 4, 2022

Test 1 is open book and open notes but with no use of Internet except to download from CRAN, R packages as needed. Smart devices such as phones must remain off. You are on your personal honor to take it without cheating. To cheat is to give or to receive answers or help; it is immoral; it vitiates recommendation letters; it ultimately destroys career development; it can get you expelled from the College; don't do it, it's not worth the consequences.

Test 1 is a form fill PDF file to open, answer, save and email to me within the **4 hour time limit** with points taken off for late submission. While taking the Test, save it as you go along, like you would any file. As discussed in class, you are to email to me the **physical file** not a cloud link and this also applies to any attached **.pdf** graph files. **Acrobat Reader DC**, for both Win and Mac, is the best way to handle form fill PDF files; get the free download: https://get.adobe.com/reader/

For this test you are informed by your trusted code from the R Editor files you've written, debugged, accumulated and curated this semester. You may also consult the R code provided by me during the semester. The last page of the test is the explanatory first page of **K funs UNIVARIATE.R** which defines useful functions as we have made use of in class.

Warning: You must store without error, the data residing on your individual test, else your answers will be wrong. Errors in data entry generate wrong answers. Incorrect or premature rounding produces wrong answers. Avoid errors in data entry, in data manipulation, in executing R code from the test, and in calling R functions.

Notice: Use your name on all files, not **Arlo**

Section	**Points**	**Information**
A. Basic Use and Operation of R for data, statistics, graphs	33	paste graph image into form or email it as **abc Arlo.pdf**
B. Critical Review of R code	5	
C. Brain weight vs. Body weight 7 vertebrate species	25	paste graph image into form or email it as **brain graph Arlo.pdf** paste **rain Arlo.R** into form
D. HIV and Age	37	paste graph image into form or email it as **HIV graph Arlo.pdf** paste **HIV Arlo.R** into form
	--------------------	

E. Extra Credit	+ 10 points max		
	Malignant Melanoma	1. Descriptive Statistics	+2 points max
		2. Graph	+6 points max
		3. Writing	+2 points max

4 hour time limit with -1 point per minute late

The simple graph has brought more information to the data analyst's mind than any other device. John Tukey, Princeton U.

A. General Use and Operation of R for data, statistics, graphs

#1-18 at 1 point each; #19 at 5 points; #20 at 2 points, #21-28 at 1 point each 33 points

The lines of R code below, simulate **Resting Heart Rate** (beats/minute) in N=98 adults who arrived for a physical examination during one week, at White Plains Hospital, NY.

From the top down, submit these 4 lines of R code to the R Console. You should use the R Editor. You must submit these lines of code exactly as given below and in top-down order, as below. Comments are not needed. **Do not email me your .R file for this problem.**

```
> set.seed(sample(1:501, 1))        # 'seed' the random number generator
> N <- 98                           # sample size
> y <- rnorm(N, 85, 7)   # N random numbers from normal distribution: mean=85, SD=7
> y <- round(y)          # round-off y to whole numbers
```

*After submitting the 4 lines of code above, you should immediately execute the code for problems 1 to 19. Once you get started with problems 1 to 19, do not go back and run any of the 4 lines above, for if you do, you will get a completely different, numeric vector **y**.*

Evaluate the ***function calls*** and the **operations** below. Enter the ***numerical results*** in the forms. ***Round to 2 decimal places***.

1. `mean(y)` **2.** `sd(y)`

3. `sd(y)/sqrt(N)` **4.** `100 * sd(y)/mean(y)`

5. `min(y)` **6.** `max(y)`

7. `skw(y)` **8.** `krt(y)` `skw()` and `krt()` are from: **`K funs UNIVARIATE.R`**

9. `mean(y) - qt(.975, N-1) * sd(y)/sqrt(N)`

10. `mean(y) + qt(.975, N-1) * sd(y)/sqrt(N)`

11. `shapiro.test(y)$p.value` **12.** `y[1:2]`

13. `hist(y, plot=FALSE)$counts`

14. `median(y)` **15.** `IQR(y)`

16. `quantile(y, .75) - quantile(y, .25)`

17. `y[N-1]`

18. `stem(y)`

Copy-paste, into the form, the **text** returned at the R Console.

Invented by John Tukey in the early 1970s a **stem plot** is a text version of a histogram.

19.

Paste **graph image** below or save/submit as **abc Arlo.pdf** but use your name not Arlo.

```
h <- hist(y, plot=FALSE)$density
dens <- density(y)

hist(y, freq=FALSE,
  main=NULL,
  ylim=range(c(0, h, dens$y)),
  border="white", col="lightblue",
  xlab="Resting Heart Rate, beats/minute",
  cex.lab=1.5, cex.axis=1.3
)

lines(dens, col="red", lwd=3) # kernel

mtext(paste("White Plains Hospital, NY",
  "\nN = 98 adults"), line=.8, cex=1.8,
  font=2, col="darkred")

mtext("red:  Kernel density", line=-1,
  adj=.98, cex=1, col="red")

box()
```

Paste **graph image** into the form or save/submit graph as **abc Arlo.pdf** if Arlo is your name else use your name. Paste image, click on it, click on a corner, then resize to fit the form.

A. General Use and Operation of R for data, statistics, graphs

#1-18 at 1 point each; #19 at 5 points; #20 at 2 points, #21-28 at 1 point each 33 points

20. Writing. Statistically describe this simulated sample of **Resting Heart Rate**. Write in a form suitable for a paper in a scientific journal. Use no more than 4 sentences. See my writing, relative to data in **PRACTICE Test 1_ANSWERS.pdf**, problem B page 4 and **Exercise 2**, problem B page 7. The first sentence is the most important as it declares what was measured and presents descriptive statistics such as N, Mean, SD and more, as appropriate.

21. Provide the single line of R code that creates your own, **personalized time stamp** along the lower right edge of the graph window using reduced font size and gray color. It should follow our conventions as to content and coding. It must to run without error in R.

No credit for partially correct code.

22. A submitted line of R code throws an error at the R Console. Provide <u>two possible reasons</u> why this happened.
You must not have any R code in your answer. Avoid vague answers. Provide a *<u>single short sentence</u>* for each.

a.

b.

23. Provide the R code you would enter at the R Console to get HTML documentation for the **describe()** function in package **psych**.

No credit for partially correct code.

24. Provide a single line of proper, R code that creates the object, **my.groups** that's a <u>character vector</u> containing 5 items.
It must run without error.

No credit for partially correct code.

25. To begin a comment in R, one inserts the character:

26. Give the name of the function that puts text in the graph window, but located outside the Cartesian coordinate system of the graph.

27. Describe the *definition*, the *practice*, and the *importance* of the "minimal call" of graph functions in R.

28. Calling a function in R, forces the R user to think about several issues. Provide 4 of these issues.

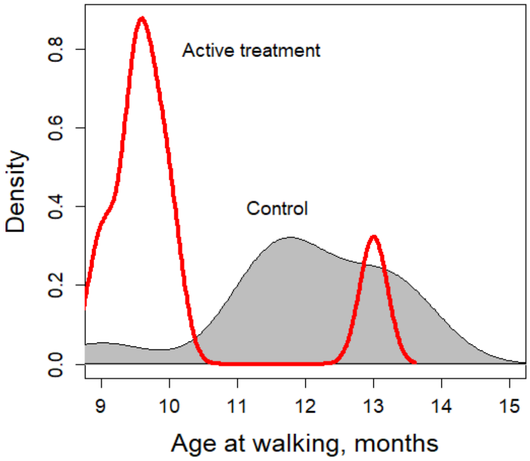
a.

b.

c.

d.

B. Critical Review of R code 5 points



The **R Editor** file named **age at walking.R** is printed below in its entirety. When submitted to the R Console, the code runs without error and returns the graph to the right. The file has 8 lines of code, plus a line of comments.

> **This code "works" but does it conform to the R coding rules and conventions we follow every class period in Bio 240?**

Provide a <u>critical review</u> of the <u>structure</u> of this file of R code, as listed below. In other words: What's wrong with the code structure and how can it be improved relative to our coding rules and conventions? Your answers must be specific and detailed. "*The code is not organized and should be made more clear*" -- is not a correct answer.

Your answers must be in sentences.

```
# age at walking.R    Martha Ubuntu, BA
active<-c(9,9.5,9.75,10,13,9.5)
control<-c(13.25,11.5,12,13.5,11.5)
d.active <- density(active)
d.control<-density(control)
plot(d.active,main="",xlim=c(9,15),xlab="Age at walking,months",cex.lab=1.6,cex.axis=1.2,col=NULL)
polygon(d.control,col="gray")
lines(d.active,col="red",lwd=4)
text(c(11.2,11.6),c(.8,.4),c("Active treatment","Control"),cex=1.2 )
```

Need 5 for full credit

1.

2.

3.

4.

5.

C. Brain weight vs. Body weight 25 points
7 vertebrate species

Mean **body weight** and mean **brain weight** were reported for various, vertebrate species (Rousseeuw, P.J. and A.M. Leroy. 1987. *Robust Regression and Outlier Detection*. Wiley, p. 57). A subset of this data is given here.

Species	Body Weight, kg *x*	Brain Weight, g *y*
Brachiosaurus	87,000	154.5
Dipliodocus	11,700	50
Triceratops	9,400	70
Asian Elephant	2,547	4,603
Giraffe	529	680
Horse	521	655
Human	62	1,320

1. **R Editor file**. Create the R Editor file, **brain Arlo.R** if your name is Arlo, else use your name. This R program is to save the above *x*,*y* data and graph it as detailed below. If I run your code it should make the exact graph you submitted. Paste into the form below, the entire contents (first line through last line) of your R Editor file. *Your program should conform to our R coding rules and conventions*. *Do not include any output. Do not include irrelevant or uncalled for code. Do not email me your .R file.* 15 points

2. Scatterplot. 10 points

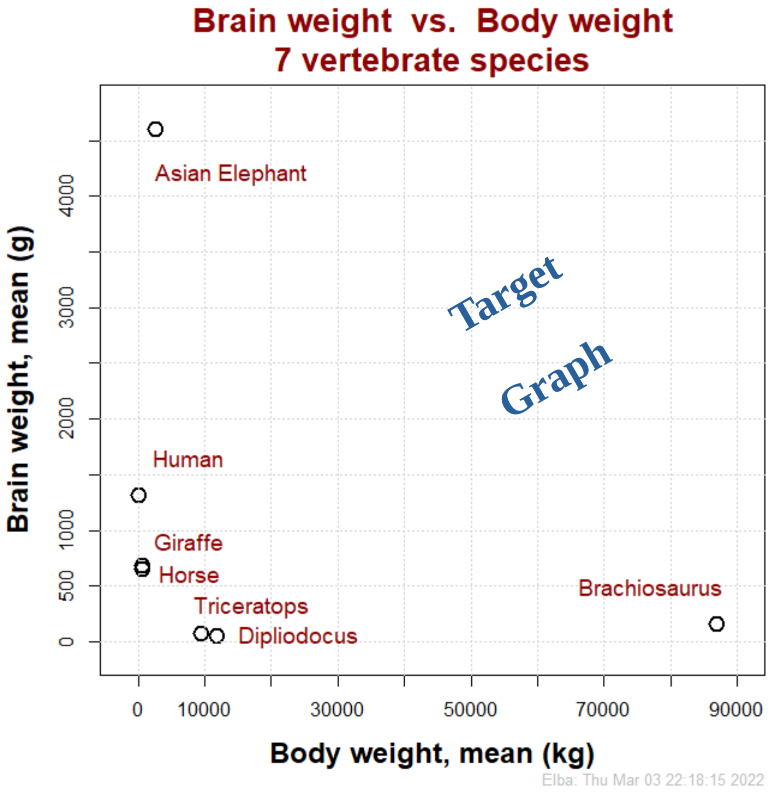
Create an enhanced, presentation quality scatterplot, matching the **Target Graph** given here. Using `text()`, label each of the 7 *x,y* points as to species. Your graph will not have the words, **Target Graph**, obviously.

In the call to `plot()`, axes must be properly labeled with units of measurement and with enlarged font size that's bold face. Note the title, the **darkred** color scheme, the enlarged plotting symbols `(cex)` and the thickness `(lwd)` of the plotting symbol `(pch=1)`.

The *x*-axis divisions every ten thousand kg and *y*-axis divisions every 500 g are by an argument new to us, `lab=c(7,7,7)`

It must have your personalized time stamp following our standards as to location, style, font size and gray color.

Paste the bit map image of the graph into the form below; click it, then click a corner to resize to fit the form **or** save graph as a .pdf file and email it to me with this test; use the name **brain graph Arlo.pdf** if your name is Arlo, else use your name.



D. HIV and Age 37 points

Find the supplied, R Editor file **HIV age Arlo.R**. Bring it up, change the name to reflect your name and add code to deliver descriptive statistics and the density histogram with kernel density estimate. On page 11 you will paste the entire contents of your .R file.
Age was recorded for N=400 HIV antibody positive men from census tracts in San Francisco in the mid 1980s during the HIV outbreak.

1. **Descriptive statistics**. Round statistics to 2 decimal places. 10 points

N = 400 Mean = SD = SEM = CV = %

skew = kurtosis = minimum = maximum =

Median = IQR =

2. **Graph**. Create an enhanced, presentation quality, density histogram of this data. See the **Target Graph** below. 10 points

Axes must be properly labeled with unit of measurement, as appropriate. Use enlarged font size for axis labels and tick labels.
From **K funs UNIVARIATE.R** the function, `mytick(2, 2, .95)` delivers the ticks as in **Target Graph**.
Histogram intervals should be 2 years wide. Note the title and the graph's color scheme.

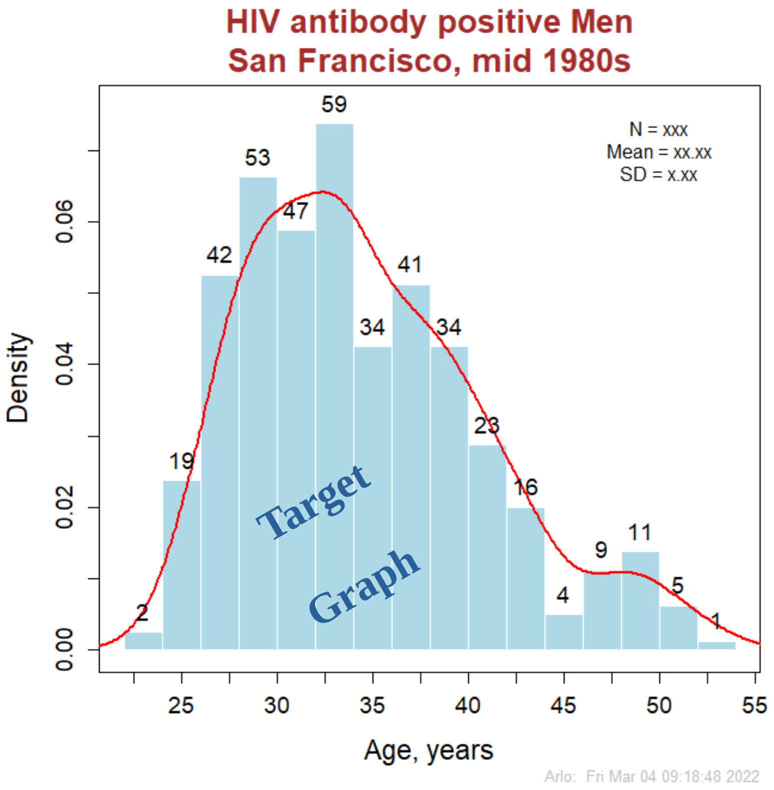
I put Frequency (not Density) atop each histogram bar. This is by a single, short call to `text()` using vectors as arguments: x, y, labels.
This is illustrated by my DEMO code block below showing for a Density histogram, how to add frequency atop bars.

```
g <- rnorm(400) # get some data

h <- hist(g, plot=F) # get the hist() object, not the graph; allows extraction
str(h) # view structure of histogram object; we will need: mids  density  count

hist(g, freq=FALSE) # make density histogram

text(h$mids, h$density+.012, labels=h$counts, cex=1.2) # try it!
# .012 is increment added to y-coordinate so
# h$counts are atop bars; change as needed
```

Put N, Mean and SD at the top right corner of graph but use real statistics, not "xx.xx" and your graph will not have **Target Graph** on it, obviously.

Your personalized time stamp is required, following our standards as to location, style, font size and color.

After the density histogram is plotted, call the `box()` function with no arguments, and notice its impact.



Paste the bit map image of your graph into the form below. Click it, then click a corner to resize to fit the form **or** save graph as a .pdf file and email it to me with this test; use the name **HIV graph Arlo.pdf** if your name is Arlo, else use your name.

3. **Writing**. Statistically describe this sample of N=400 ages for HIV positive men in San Francisco, CA in the mid 1980s. Write in a form suitable for a paper in a scientific journal. Use no more than 4 sentences. See my writing samples. The first sentence is the most important as it declares what was measured and presents descriptive statistics such as N, Mean, SD and more, as appropriate. 2 points

4. **R Editor file**. Paste the entire contents of your R Editor file for this problem, e.g., **HIV age Arlo.R** if your name is Arlo. 15 points
If I run your code it should make the exact graph you submitted. Paste into the form below, the entire contents (first line through last line) of your R Editor file. *Your program should conform to our R coding rules and conventions*. *Do not include any output. Do not include irrelevant or uncalled for code.* ***Do not email me your .R file***.

E. Extra Credit + 10 maximum

Malignant Melanoma. In package **ISwR** we have the **melanom** data set. This is a 205 x 7 column data frame on patients after a malignant melanoma operation by Dr. K.T. Drzewiecki at Odense University Hospital in Denmark.

```
# code to get you started

library(ISwR) # load from your drive, the package into R Console
data(melanom) # load the data

str(melanom)
#'data.frame':   205 obs. of  6 variables:
# $ no    : int  789 13 97 16 21 469 685 7 932 944 ...
# $ status: int  3 3 2 3 1 1 1 1 3 1 ...
# $ days  : int  10 30 35 99 185 204 210 232 232 279 ...
# $ ulc   : int  1 2 2 2 1 1 1 1 1 1 ...
# $ thick : int  676 65 134 290 1208 484 516 1288 322 741 ...
# $ sex   : int  2 2 2 1 2 2 2 2 1 1 ...

# status:  1 is dead from melanoma, 2 is still alive, 3 is dead from other causes
# days:    observation time
# thick:   tumor thickness, 1/100 mm
# sex:     1 female, 2 male
```

Here, we are interested in **tumor thickness** for those **alive** versus those who **died** from melanoma.

```
# ----------------------------------
#    tumor thickness for dead v. alive   -----------------------------
# ----------------------------------

melanom$thick <- melanom$thick * .01  # convert to mm

# -------------------------------------------------------------------
dead  <- subset(melanom, status == 1)  # extract those 'dead' from melanoma
str(dead)  # view result  N=57

#'data.frame':   57 obs. of  6 variables:
# $ no    : int  21 469 685 7 944 558 2 233 418 777 ...
# $ status: int  1 1 1 1 1 1 1 1 1 1 ...
# $ days  : int  185 204 210 232 279 295 386 426 469 529 ...
# $ ulc   : int  1 1 1 1 1 1 1 1 1 1 ...
# $ thick : num  12.08 4.84 5.16 12.88 7.41 ...
# $ sex   : int  2 2 2 2 1 1 1 2 1 2 ...

# dead <- melanom[melanom$status ==1, ]  # explicit indexing method
# alive <- melanom[melanom$status ==2, ] #

alive <- subset(melanom, status == 2)  # extract those still 'alive'
str(alive) # view result  N=134        # after 15.2 years

#'data.frame':   134 obs. of  6 variables:
# $ no    : int  97 455 29 636 10 468 130 808 390 802 ...
# $ status: int  2 2 2 2 2 2 2 2 2 2 ...
# $ days  : int  35 1499 1508 1510 1512 1542 1557 1563 1605 1627 ...
# $ ulc   : int  2 2 1 2 2 2 2 2 2 2 ...
# $ thick : num  1.34 1.29 8.38 1.94 0.16 0.16 1.29 0.32 1.13 1.62 ...
# $ sex   : int  2 2 2 1 1 1 1 1 1 1 ...
```

Now we have 2 data frames -- dead and alive

```
# get kernel density for tumor thickness for 'alive' and for 'dead'

dead.dens  <- density(dead$thick, from=min(dead$thick), to=max(dead$thick))
alive.dens <- density(alive$thick, from=min(alive$thick), to=max(alive$thick))
```

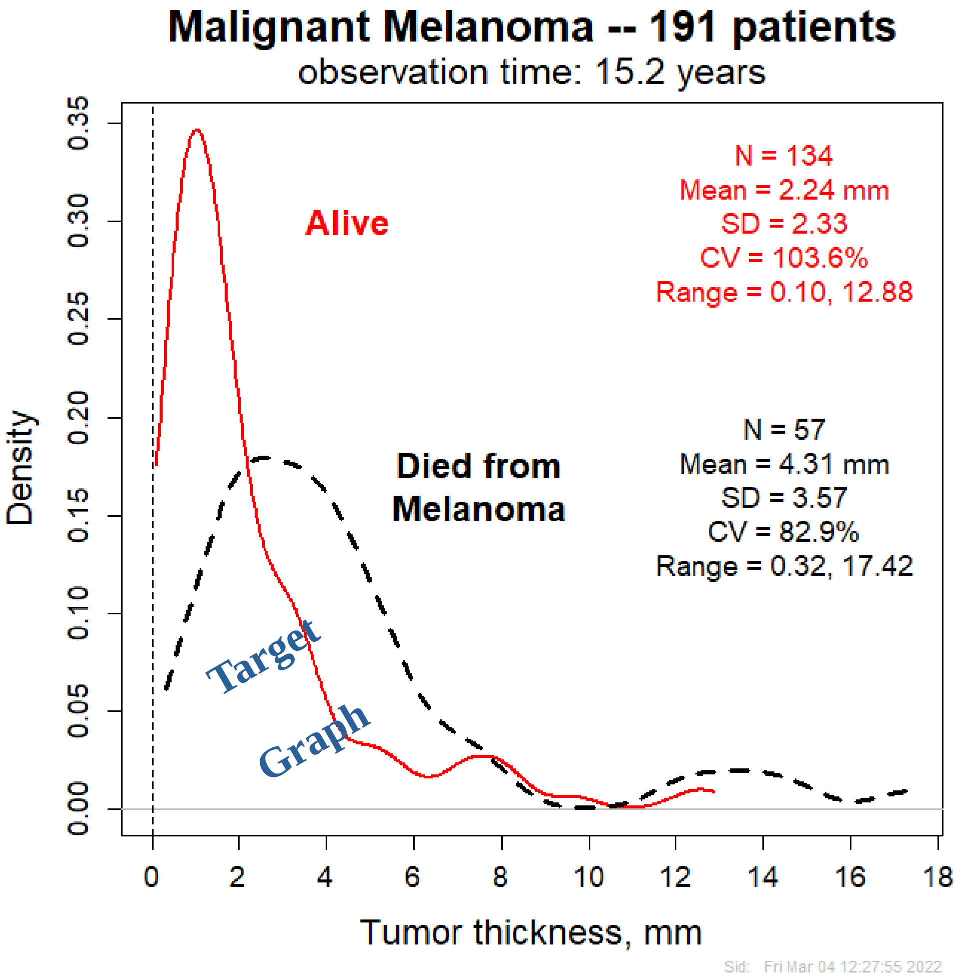
```
plot(alive.dens, main="") # minimal call for 'dead'
lines(dead.dens)          # add 'dead'
abline(v=0, lty=2)


# -----------
# final graph -------------------------- Spring 2022  -----
# -----------

# add your code here, building on the minimal calls above




# ----------------------
# Descriptive statistics -----------------------
# ----------------------

# getting started

psych::describe(alive$thick)

psych::describe(dead$thick)
```

1. **Descriptive Statistics**. Develop a set of descriptive statistics for tumor thickness for the Alive group and for the Died group of patients. Roundoff to 2 decimals. + 2 points maximum

Tumor Thickness, mm	Alive	Died
N		
Mean		
SD		
CV%		
SEM		
Range		
95% CI of Mean		
skewness		
kurtosis		
Shapiro-Wilk test p-value		

2. Graph + 6 maximum

Create a graph resembling my Target Graph below. Save it as **tumor Arlo.pdf** if your name is Arlo, else use your name. Email this .pdf graph file along with Test 2.



3. Writing. + 2 maximum

Statistically describe tumor thickness in those who survived versus those who died from melanoma. Write in a form suitable for a paper in a scientific journal. Make use of descriptive statistics and graphical visualization.

```
# K funs UNIVARIATE.R    February 18, 2022    Author: Dwight Kincaid, PhD
#                                          dwight.kincaid@lehman.cuny.edu
#
#  SOURCE this file; its new funs can then be called. Some funs call
#  other funs defined in this file. Missing values in the data are OK.
#
#  These funs only deal with UNIVARIATE data. Commented-out DEMO CODE after
#  each fun definition should be examined and run. If printed it's 24 pages.
#
#  DEMO DATA includes the N=298, data(IgM) in library(ISwR) as well as
#  simulated data by sampling from theoretical probability distributions.
#
#  I use base R funs as much as possible, to reduce reliance on contributed
#  CRAN packages for reasons, including code transparency for students and
#  for code 'survival' albeit at slightly slower speed for computationally
#  intensive tasks although this is trivial unless N is large.
# ---------------------------------------------------------------------------
##
##  FUNCTION  DEFINITIONS  --  for use on a single, numeric sample: a vector
##                                                  missing values OK
#      NEW FUNCTIONS
#
#   1. freq.table(y, roundoff, ...)        # frequency distribution table
#
#   2. EDA(y, data.name)                   # 4 EDA graphs, heavily annotated
#
#   3. CV.percent(y)                       # coefficient of variation as %
#
#   4. classical.CI.mean(y)            # 90,95,99% CI of mean by normal theory
#   5. boot.CI.mean(y, NS, data.name) # nonparametric bootstrap CI of mean
#
#   6. my.stats(y)                                  # descriptive stats
#   7. new.stats(y, roundoff, data.name, norm.test) # descriptive stats & more
#
#   8. normal.QQ.plot(y)                 # graphical assessment
#   9. Normal.kernel.band(y, NS=1000)    #    of normality based on
#                                        #       simulation from N(n, mean, sd)
#
#  10. %mc.skew%  # a binary operator: Monte Carlo sim test for SKEW     Ho: skw=0
#  11. %mc.kurt%  # a binary operator; Monte Carlo sim test for KURTOSIS Ho: krt=0
#
#  12. %boot.skew%  # a binary operator: nonparametric bootstrap CI for SKEW
#  13. %boot.kurt%  # a binary operator: nonparametric bootstrap CI for KURTOSIS
#
#  14. Tukey.outliers(y) # identify outliers by Tukey's 1.5 IQR rule in boxplot()
#
#  15. %bootMedian95CI%  # a binary operator; percentile bootstrap CI of median
#
#  16. skw(y)        # skewness (type 3); Kincaid code in base R
#  17. skw2(y)       # same as skw() but faster
#
#  18. krt(y)        # kurtosis (type 3); Kincaid code in base R
#  19. krt2(y)       # same as krt() but faster
#
#  --  DEMO functions --
#
#  20. DEMO.What.is.SEM()   21. DEMO.SD.innate.biol.variability
#
#  --  purely convenience functions  --
#  xx. thick.line(N)      # draw line of "="
#  xx. thin.line(N)       # draw line of "-"
#  xx. my.pause()         # pause between graphs
#  xx. mytick(nx=2, ny=2, tick.ratio=0.5)  # minor ticks to base R graphs
#  --  functions to add  --
#  xx. myBanner     xx. FooterHeader  xx. BIG DATA vectors
#  xx. leaveOneOut  xx. simUnivar     xx. runs tests
#  xx. GIF graphs   xx. animations    xx. more EDA    xx. Bayesian funs
```

--- END of Test 1: Bio 240. *Biostatistics*. Spring 2022 ---