Defining the Problem

Bootstrapping is a statistical resampling distribution method used to gather an unbiased statistic from a large enough parent distribution. It was introduced in 1979 by Bradley Efron as an extension to an older resampling method known as the "jackknife". Although its coding is relatively simple compared to other sampling methods, it does require the use of multiple statistical techniques such as mean sampling and histogram dynamics.

General Process of Bootstrapping:

1. A quantitative parent distribution/sample must be present before bootstrapping takes place. This sample can have values ranging from negative to positive infinity and theoretically can have a sample size of infinity.
2. The list is then randomly sorted from the range of the parent distribution. There can be as many random lists based on the parent distribution as the bootstrapper chooses. Each randomly generated list, however, must have the same sample size as the parent distribution. The number of random child distributions is equal to the number of iterations in bootstrapping.
3. The bootstrapper then chooses to take a statistical value from each random child distribution. Whichever statistical value is to be found must be the same for every random distribution created.
4. Once each statistical value has been found from each random child distribution, the statistical values are then plotted on a histogram with a frequency of one.
5. According to the central limit theorem and the rules of sampling proportion, the histogram should form a Gaussian shape- which indicates a normal distribution- if the number of iterations satisfies the conditions of the theorem. This is the basis of bootstrapping.
6. Since the distribution is based on a sample mean, a further practical use would be to calculate the differences in the sample mean/proportion to the known population mean or parameter.

As the study of statistics continues, bootstrapping is in more of a need to be readily accessible to students and statisticians alike. The problem is finding a readily accessible device capable of performing this process.

Specifications

1. The program will start up on command and through the previous entry command.
2. It will utilize all the default available lists and matrices of the TI device.
3. It will create as many random child distributions and support any parent distributions as long as it does not exceed the calculator's limitations.
4. It will properly exit through prompt or through a interrupted runtime.
5. It will present no errors during runtime (with the exception of the break error due to user interference).
6. It will go through the general process of bootstrapping through the use of the statistical value of means and sample means.
7. It will display the histogram with the expected results through the central limit theorem.
8. The user can choose whether to delete all data created by program or not in order to view the data outside of the program.

Programming Language used: TI-BASIC
Device used to program: TI-84 Plus CE (Texas Instruments)
Total number of lines (without comments): 433
Notes:

- Commands involving parentheses and quotations do not need end quotations and/or parentheses.
- In TI-BASIC, comments are marked with a :"[comment]" next to the command (quotations included; brackets excluded).
- By default, the calculator does not allow full lowercase typing unless through the use of a computer or system app that allows access.
- The TI-84 Plus CE/CSE supports a 26x10 character display matrix. The code below will not be displayed correctly if executed with non-color TI graphing calculators.
- Labels can be up to two characters long, from 00 to ZZ.

Bibliography:

The following sources was used for the research of bootstrapping:

Hand, Nick. "Bootstrap Resampling." *YouTube*. YouTube, 13 Dec. 2012. Web. 20 Mar. 2016. <https://www.youtube.com/watch?v=ZCXg64l9R_4>.

Taylor, Courtney. "What Is Bootstrapping in Statistics?" *About.com Statistics*. About, 16 Dec. 2014. Web. 20 Mar. 2016. <http://statistics.about.com/od/Applications/a/What-Is-Bootstrapping.htm>.

https://en.wikipedia.org/wiki/Bootstrapping_(statistics)

Source Code:

```
PROGRAM:BOOTSTRP
ClrHome
rand
Output(1,1,"-------------------------
Output(2,1,"BOOTSTRAPPING PROGRAM    "
Output(3,1,"-------------------------
Output(5,1,"BY NEELOY GOMES
Output(7,1,"A RESAMPLING DISTRIBUTION
Output(8,1,"PROGRAM
Output(10,1,"PRESS ENTER TO CONTINUE
Pause
Lbl 99
ClrHome
Menu("MAIN MENU:","USE USER SAMPLE",1,"USE GIVEN
SAMPLE",2,"EXIT",3)
Lbl 1
ClrHome
Menu("WHICH LIST HAS
SAMPLE?","L1",11,"L2",12,"L3",13,"L4",14,"L5",15,"L6",16,"BACK...",99)
Lbl 2
Menu("WHICH TYPE OF SAMPLE?","SYMMETRIC UNIMODAL",C5,"SYMMETRIC
BIMODAL",C6,"RIGHT-SKEWED",C7,"LEFT-
SKEWED",C8,"UNIFORM",C9,"BACK...",99)
Lbl C5
1→M
ClrHome
1→E
50→F
Goto D2
Lbl C6
2→M
ClrHome
1→E
50→F
Goto D2
Lbl C7
3→M
ClrHome
1→E
```

```
50→F
Goto D2
Lbl C8
4→M
ClrHome
1→E
50→F
Goto D2
Lbl C9
0→M
ClrHome
1→E
50→F
Lbl D2
ClrHome
Menu("OVERWRITE WHICH LIST?","L₁",21,"L₂",22,"L₃",23,"L₄",24,"L₅",25,"L₆",26)
End
Lbl 21
Disp "OVERWRITING LIST..."
{0}→L₁
(E+F)/2→U
If M=1
Then
iPart(randNorm(U,√((F−U)²+(E−U)²),30))→L₁
End
If M=2
Then
iPart(randNorm(U,√((F-U)²+(E-U)²),30))→ʟB
iPart(randNorm(U,√((F-U)²+(E-U)²),30))→ʟC
augment(ʟB,ʟC)→L₁
End
If M=3
Then
iPart(E+Frand(30))→L₁
End
If M=4
Then
iPart(E+F√(rand(30)))→L₁
End
If M=0
```

```
Then
randInt(E,F,30)→L₁
End
1→B
Goto C1
Lbl 22
Disp "OVERWRITING LIST..."
{0}→L₂
(E+F)/2→U
If M=1
Then
iPart(randNorm(U,√((F−U)²+(E−U)²),30))→L₂
End
If M=2
Then
iPart(randNorm(U,√((F-U)²+(E-U)²),30))→ʟB
iPart(randNorm(U,√((F-U)²+(E-U)²),30))→ʟC
augment(ʟB,ʟC)→L₂
End
If M=3
Then
iPart(E+Frand(30))→L₂
End
If M=4
Then
iPart(E+F√(rand(30)))→L₂
End
If M=0
Then
randInt(E,F,30)→L₂
End
2→B
Goto C1
Lbl 23
Disp "OVERWRITING LIST..."
{0}→L₃
(E+F)/2→U
If M=1
Then
iPart(randNorm(U,√((F−U)²+(E−U)²),30))→L₃
```

```
End
If M=2
Then
iPart(randNorm(U,√((F-U)²+(E-U)²),30))→ʟB
iPart(randNorm(U,√((F-U)²+(E-U)²),30))→ʟC
augment(ʟB,ʟC)→L₃
End
If M=3
Then
iPart(E+Frand(30))→L₃
End
If M=4
Then
iPart(E+F√(rand(30)))→L₃
End
If M=0
Then
randInt(E,F,30)→L₃
End
3→B
Goto C1
Lbl 24
Disp "OVERWRITING LIST..."
{0}→L₄
(E+F)/2→U
If M=1
Then
iPart(randNorm(U,√((F−U)²+(E−U)²),30))→L₄
End
If M=2
Then
iPart(randNorm(U,√((F-U)²+(E-U)²),30))→ʟB
iPart(randNorm(U,√((F-U)²+(E-U)²),30))→ʟC
augment(ʟB,ʟC)→L₄
End
If M=3
Then
iPart(E+Frand(30))→L₄
End
```

```
If M=4
Then
iPart(E+F√(rand(30)))→L₄
End
If M=0
Then
randInt(E,F,30)→L₄
End
4→B
Goto C1
Lbl 25
Disp "OVERWRITING LIST..."
{0}→L₅
(E+F)/2→U
If M=1
Then
iPart(randNorm(U,√((F−U)²+(E−U)²),30))→L₅
End
If M=2
Then
iPart(randNorm(U,√((F-U)²+(E-U)²),30))→ʟB
iPart(randNorm(U,√((F-U)²+(E-U)²),30))→ʟC
augment(ʟB,ʟC)→L₅
End
If M=3
Then
iPart(E+Frand(30))→L₅
End
If M=4
Then
iPart(E+F√(rand(30)))→L₅
End
If M=0
Then
randInt(E,F,30)→L₅
End
5→B
Goto C1
Lbl 26
Disp "OVERWRITING LIST..."
```

```
{0}→L6
(E+F)/2→U
If M=1
Then
iPart(randNorm(U,√((F−U)²+(E−U)²),30))→6
End
If M=2
Then
iPart(randNorm(U,√((F-U)²+(E-U)²),30))→ᴸB
iPart(randNorm(U,√((F-U)²+(E-U)²),30))→ᴸC
augment(ᴸB,ᴸC)→L6
End
If M=3
Then
iPart(E+Frand(30))→L6
End
If M=4
Then
iPart(E+F√(rand(30)))→L6
End
If M=0
Then
randInt(E,F,30)→L6
End
6→B
Goto C1
Lbl C1
ClrHome
Disp "THE LIST HAS BEEN","OVERWRITTEN."
Pause
ClrHome
Menu("NEXT STEP?","BOOTSTRAP IMMEDIATELY",1,"CHECK THE SAMPLE
CREATED",C2)
Lbl C2
ClrHome
Disp "CHECK THE SAMPLE BY GOING","TO STAT AND EDIT."," "
Disp "RESTART THE ","PROGRAM TO USE THE SAMPLE."
Stop
End
Lbl 3
```

```
Menu("SAVE GRAPH AND LIST DATA?","YES",4,"NO",0)
Lbl 4
ClrHome
Disp "GRAPHS AND LISTS SAVED FOR","FURTHER VIEWING."," ","PRESS
ENTER TO CLOSE THE","PROGRAM AND ZOOMSTAT TO","DISPLAY THE
GRAPH."
AxesOn
Pause
ClrHome
Stop
Lbl 0
PlotsOff
ClrDraw
DelVar ⌞A
DelVar ⌞B
DelVar ⌞C
DelVar ⌞BOOT
DelVar ⌞FREQ
DelVar ⌞θ
-10→Xmin
10→Xmax
-10→Ymin
10→Ymax
ClrDraw
ClrHome
Disp "GRAPHS AND LISTS CREATED","BY THIS PROGRAM HAVE
BEEN","ERASED."
AxesOn
Stop
Lbl 11
If dim(L1)=0
Then
ClrHome
Disp "THERE IS NO DATA IN THIS","SAMPLE.
Pause
Goto 99
Else
L1→A
dim(L1)→L
```

```
1→B
Goto A1
End
Lbl 12
If dim(L₂)=0
Then
Disp "THERE IS NO DATA IN THIS","SAMPLE.
Pause
Goto 99
Else
L₂⟶A
dim(L₂)⟶L
2→B
Goto A1
End
Lbl 13
If dim(L₃)=0
Then
Disp "THERE IS NO DATA IS THIS","SAMPLE.
Pause
Goto 99
Else
L₃⟶A
dim(L₃)⟶L
3→B
Goto A1
End
Lbl 14
If dim(L₄)=0
Then
Disp "THERE IS NO DATA IN THIS","SAMPLE.
Pause
Goto 99
Else
L₄⟶A
dim(L₄)⟶L
4→B
Goto A1
End
Lbl 15
```

```
If dim(L5)=0
Then
Disp "THERE IS NO DATA IN THIS","SAMPLE.
Pause
Goto 99
Else
L5⟶A
dim(L5)⟶L
5→B
Goto A1
End
Lbl 16
If dim(L6)=0
Then
Disp "THERE IS NO DATA IN THIS","SAMPLE.
Pause
Goto 99
Else
dim(L6)⟶L
L6⟶A
6→B
Goto A1
End
Lbl A1
{0}→ᴸBOOT
{0}→ᴸθ
ClrHome
Disp "RESAMPLE SIZE: ","(1-999)"
Prompt R
If R<1 or R>999
Then
Output(5,1,"THE CALCULATOR CANNOT
Output(6,1,"HANDLE THIS SAMPLE SIZE.
Output(7,1,"PLEASE INPUT A SAMPLE SIZE
Output(8,1,"WITHIN RANGE."
Pause
Goto A1
Else
Disp "PLEASE WAIT..."
R→dim(ᴸθ)
```

```
End
Lbl 98
Disp " ","HOW MANY ITERATIONS?","(1-999)
Prompt I
If I<1 or I>999
Then
ClrHome
Disp "THE CALCULATOR CANNOT","HANDLE THIS
MANY","ITERATIONS.","PLEASE INPUT A NUMBER","WITHIN RANGE."
Pause
ClrHome
Goto 98
Else
Disp "PLEASE WAIT..."
I→dim(ʟBOOT)
End
1→X
Lbl 50
ClrHome
Output(1,1,"SAMPLE LIST:
If B=1
Then
Output(1,14,"L₁
End
If B=2
Then
Output(1,14,"L₂
End
If B=3
Then
Output(1,14,"L₃
End
If B=4
Then
Output(1,14,"L₄
End
If B=5
Then
Output(1,14,"L₅
End
```

```
If B=6
Then
Output(1,14,"L6
End
Output(2,1,"SAMPLE SIZE:
Output(2,14,L
Output(3,1,"ITERATIONS:
Output(3,13,I
Output(4,1,"RESAMPLE SIZE:"
Output(4,16,R
Output(6,1,"ARE YOU SATISFIED WITH
Output(7,1,"THESE SETTINGS?"
Pause
Menu("SATISFIED?","YES",70,"NO",99)
Lbl 70
ClrHome
Disp "BOOTSTRAPPING..."
1→Y
Output(3,1,"THIS MAY TAKE SEVERAL
Output(4,1,"MINUTES DEPENDING ON THE
Output(5,1,"SAMPLE SIZE AND THE NUMBER
Output(6,1,"OF ITERATIONS.
Output(9,1,"ITERATIONS COMPLETE:
Output(10,4,"/"
Output(10,5,I
Output(10,1,"0"
While Y≤I
1→Z
While Z≤R
randInt(1,L)→K
ʟA(K)→ʟθ(Z)
Z+1→Z
End
mean(ʟθ)→ʟBOOT(Y)
Output(10,1,Y
Y+1→Y
End
ClrDraw
AxesOn
Plot1(Histogram,ʟBOOT,1,BLUE)
```

```
Lbl 5
ZoomStat
DispGraph
Pause
ClrHome
Menu("START OVER?","YES",99,"NO",3,"DISPLAY GRAPH AGAIN",5)
```