

Sampling Distributions

Calculator Note 7A: Generating Sampling Distributions

Many statistics computer programs efficiently perform sampling from data sets and offer the option of sampling with and without replacement. Parallel programs for the TI-83 Plus and TI-84 Plus can be written and executed but frequently without the efficiency and control of statistics computer programs.

What follows is a procedure that generates a sampling distribution without requiring a program. This example uses the `randNorm()` command to create samples from a normally distributed population. You can also use the `randInt()` command or the `randBin()` command to create three uniformly distributed populations or binomially distributed populations, respectively. All three commands are found by pressing **MATH** and looking in the PRB menu.

This example takes 50 random samples of size 5 from a normal distribution with mean 100 and standard deviation 12. Then the sample means are calculated and displayed.

- Enter `randNorm(100,12,50)→L1` to load 50 randomly selected numbers from the population into list L1.

```
randNorm(100,12,
50)→L1
```

L1	L2	L3	1
84.034			
112.61			
99.558			
102.86			
92.507			
120.47			
L1(1)=80.97148452...			

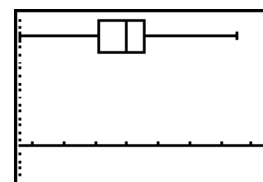
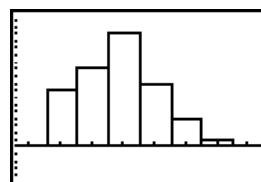
- Repeat step a for lists L2, L3, L4, and L5.
- Each row of the List Editor screen constitutes a random sample of size 5. Define list L6 as the sum of the rows divided by 5.

L4	L5	M1	6
99.344	106.85	100.43	
105.39	114.67	104.42	
93.706	114.24	104.74	
78.503	113.98	100.05	
82.759	98.832	95.472	
95.423	108.49	101.26	
93.788	116.15	108.47	
L6 =...L3+L4+L5)/5"			

List L6 now contains 50 sample means from a normal population with mean 100 and standard deviation 12.

- Analyze the sampling distribution in list L6 both numerically and graphically.

```
1-Var Stats
x=100.0762579
Σx=5003.812895
Σx²=502371.287
sx=5.729296269
σx=5.67171394
n=50
```



[84.89, 121.49, 4.58, -4.81, 18.72, 1]

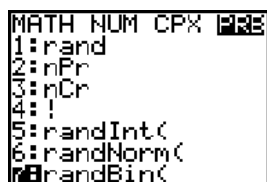
[84.89, 121.49, 4.58, -4.81, 18.72, 1]

Needless to say, a greater number of samples would produce a more accurate picture of the sampling distribution.

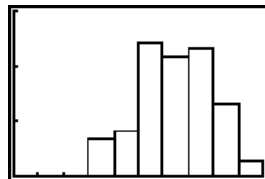
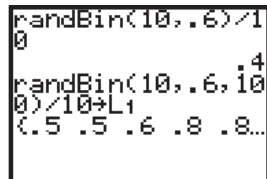
Calculator Note 7B: Activity 7.3a–Buckle Up!

On the TI-83 Plus and TI-84 Plus, the random binomial command, `randBin(`, generates a random integer from a specified binomial distribution. A binomial distribution counts the number of successes for a success-or-failure probability experiment, so you can use `randBin(` to create sampling distributions for this activity.

You find the `randBin(` command by pressing `MATH`, arrowing over to `PRB`, and selecting `7:randBin(`. The syntax of the command is `randBin(sample size, probability of success, number of samples)`. If the number of samples is not specified, the calculator assumes 1.



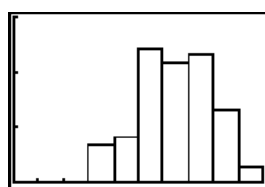
For example, to select a random sample of size 10 from a population with 0.6 probability of success and count the number of successes, enter `randBin(10,.6)` into the Home screen. To calculate the sample proportion, divide the result by 10, or `randBin(10,.6)/10`. To do 100 samples and store the sample proportions in list `L1`, enter `randBin(10,.6,100)/10>L1`. You can now create a histogram of list `L1`. When setting the window, use `Xscl = 1/sample size`.



[0, 1, 0.1, 0, 30, 10]

If you want to create a relative frequency histogram, similar to those shown in Display 7.40 on page 449 in the student book, run the `FREQTABL` program (see Calculator Note 0J), which puts data values in list `L2` and frequencies in list `L3`. Then define list `L4` with the expression `L3/100`. Now create a relative frequency histogram using `L2` for `Xlist` and `L4` for `Freq`. Be sure to adjust `Ymin`, `Ymax`, and `Yscl`.

L2	L3	L4	#
3	7	.07	
4	8	.08	
5	24	.24	
6	22	.22	
7	23	.23	
8	13	.13	
9	3	.03	
L4 = "L3/100"			



[0, 1, 0.1, 0, 0.3, 0.1]

Follow a similar procedure to do sampling distributions for $n = 20$ and $n = 40$. Change the value of `Xscl` accordingly. If you use relative frequency histograms, you'll need to run `FREQTABL` for each new distribution and redefine `L4` (unless you use a dynamic definition).

Calculator Note 7C: Demonstrating the Central Limit Theorem, and Simulating Sampling from a Population—The SAMPMEAN Program

To demonstrate the Central Limit Theorem for a Sample Mean, you can use a calculator procedure similar to the one described in Calculator Note 7A. Look back at the first screen in part d. Notice that for a population that was normally distributed with mean 100 and standard deviation 12, fifty samples gave $\mu_{\bar{x}} = 100.076$ and $\sigma_{\bar{x}} = 5.729$.

These experimental values are close to the theoretical values dictated by the Central Limit Theorem, $\mu_{\bar{x}} = \mu = 100$ and $\sigma_{\bar{x}} = \sigma \sqrt{n} = 5.367$.

To demonstrate the Central Limit Theorem for the Sample Mean from a skewed population, you can perform the same procedure on a highly skewed population, such as a binomial distribution with $p = 0.20$. (See Calculator Note 4A.)

The SAMPMEAN Program

As an alternative to the sampling procedure, you can use the program SAMPMEAN, which selects random samples (with replacement) from a population stored in list L1, displays a sampling distribution, and calculates the mean of the sampling distribution and the standard error of the mean. The program is slow, but is valuable for demonstrating the Central Limit Theorem for both normal and non-normal distributions.

Before running the program, enter the population data into list L1. Run the program by pressing **PRGM** and selecting SAMPMEAN from the EXEC menu. At the prompts, enter the number of samples, S, and the sample size, N. Press **ENTER** after each value. The program collects samples and stores the sample means in list L2. A menu will appear from which you can view the statistics for the sampling distribution (mean and standard error), the statistics for the population data (mean and standard deviation), a histogram of the sampling distribution, or a histogram of the population. After viewing any of the statistics or histograms, press **ENTER** to return to the menu. To collect a new sampling distribution, choose 5:AGAIN. To end the program, choose 6:END.

L1	L2	L3	1
-----	-----	-----	
W-----			

L1(1)=2			

SAMPLE MEANS
NUM SAMPLES? 50
NUM IN EACH? 10

CHOOSE:
1:STATS MEANS
2:STATS DATA
3:HIST MEANS
4:HIST DATA
5:AGAIN
6:END

NUM IN EACH	10
MEAN	2.112
SD	.3114416773

```

PROGRAM: SAMPMEAN
PlotsOff :FnOff :ClrHome
Disp "SAMPLE MEANS"
Disp ""
Lbl C
ClrList L2,L3
1-Var Stats L1
Input "NUM SAMPLES?",S
Input "NUM IN EACH?",N
For(J,1,S,1)
For(I,1,N,1)
L1(randInt(1,dim(L1)))→L3(I)
End
mean(L3)→L2(J)
ClrList L3
End
min(L1)-2→Xmin
max(L1)+2→Xmax
(max(L1)-min(L1))/10→Xscl
Lbl E
ClrHome
Menu("CHOOSE:", "STATS MEANS",
A, "STATS DATA", F, "HIST MEANS",
B, "HIST DATA", G, "AGAIN",
C, "END", D)
Lbl A
ClrHome
Disp "NUM IN EACH",N
Disp "MEAN",mean(L2)
Disp "SD",stdDev(L2)

Pause
Goto E
Lbl B
PlotsOff
Plot1(Histogram,L2)
S/2→Ymax
-.5→Ymin
1→Yscl
Xscl/
2→Xscl
DispGraph
Pause
Xscl*2→Xscl
Goto E
Lbl F
ClrHome
Disp "DATA"
Disp "MEAN", $\bar{x}$ 
Disp "SD", $\sigma_x$ 
Pause
Goto E
Lbl G
PlotsOff
Plot1(Histogram, L1)
dim(L1)/2→Ymax
-.5→Ymin
1→Yscl
DispGraph
Pause
Goto E
Lbl D

```