Lab Exercise 1 9 Sep 02

1. Consider the sample of values 1, 2, 6, 8, 20.
   (a) Compute the mean.
   (b) Recompute the mean after adding 10 to each value.
   (c) Recompute the mean after multiplying each value by 2.
   (d) Notice what happened.
2. Compute the variance and standard deviation doing the following:
   (a) Use the original values.
   (b) Recompute both after adding 10 to each value.
   (c) Recompute both after multiplying each value by 2.
   (d) Notice what happened.
3. Change the last value separately to 50, 200, 2000 and recompute the mean.
4. Change the first value to .1, .01, .001, .0001 and recompute the mean, otherwise using the original sample.
5. Compute the geometric mean of the following:
   (a) The original data.
   (b) After changing the last value to 50, 200, 2000.
   (c) After changing the first value to .1, .01, .001, .0001, otherwise using the original sample.
6. Generate a population of 20000 values from a uniform $[0, 100]$ distribution. If you want to use the same random numbers as other students, first run `set.seed(123)`.
7. Compute the *population* mean.
8. Use the `sample` function to pull a random sample of size 10 from this population without replacement. Compute the mean of this random sample.
9. Repeat the last step 10 times and examine the variation of these sample means.
10. Generate a sample of 20 random normally distributed values having population mean 50 and standard deviation 10. Compute the sample mean and standard deviation.
11. (Extra): Put these 20 values along with two more variables also of length 20 to be generated as below, into a data frame. Print the data frame.
    (a) A sample from the numbers 1-100 with replacement.
    (b) A sample from the values `'cat','dog'` with replacement.

Lab Exercise 2 23 Sep 02

1. Issue the command `search()` and study the output, especially what's in the first few search positions. **Repeat this process** after the next problem and after **each** time you issue an `attach` or `detach` command.

2. The latest version of the Hmisc library is already installed in the lab. Make S-PLUS have access to the functions and help files in this library.

3. S-PLUS comes with many data frames built-in that don't even need to be imported. Click on the `Help` menu and select `Language Reference`. Show `Topics` and get to `Datasets`. Find the entry for the `lung` dataset, which may be under the heading `survival.datasets`. Look at the description of the variables. Issue the command `find(lung)` and note that the data frame is found because it is in a directory that is in the search list.

4. Use the Hmisc `describe` function to find out about the data. Put the function's output in a new window. Note that the `inst` (institution) variable is not handled correctly. Explain why.

5. Run some of the graphical functions in Hmisc for inspecting the data.

6. Note that the variables do not have labels and the `sex` variable is poorly documented. Transform `sex` into a self-documenting categorical variable and give `ph.ecog` a descriptive label attribute. Make S-PLUS recognize `inst` as a truly categorical variable (you do not need to define `levels` as we do not know them) and provide a label for one of the other variables that needs one. Save the modified data frame under a new name such as `lung2` until you are confident that the operations were done correctly (you can check this with `describe` on the new data frame which will take advantage of the new labels and level information).

7. Get easy access to all the variables in the data frame so that they may be referenced without using a prefix. Show where the `age` and `meal.cal` variables are being found now.

8. Compute the mean `wt.loss` stratified by `sex`. As `wt.loss` has some `NA`s you will need to add `na.rm=T` as a final argument to `tapply`.

9. Remove access to the individual variables in `lung` and gain access to the individual variables but only for males. Find where `age` is coming from now.

Lab Exercise 3 14 or 21 Oct 02

1. Simulating repeated samples of a given sample size is a good way to understand sampling properties of statistics such as the mean. The most concise way to generate m random samples each of size n at one time is to simulate a matrix with n rows and m columns. For example you can generate 100 samples of 50 observations each from the standard normal distribution using

   ```
   m ← 100
   n ← 50
   x ← matrix(rnorm(n*m), ncol=m)
   ```

   Then, to separately compute means for each of the 100 samples one can tell the `apply` function to compute column means:

   ```
   means ← apply(x, 2, mean)   # vector of length 100
   ```

   If one had only the luxury of a single sample, say the first column of `x`, one could estimate the population mean by the sample mean of this column (`x[,1]`) and then estimate the precision of the mean using the standard error of the mean, after computing the standard deviation of the vector `x[,1]`. Generate 100 samples of size 50 as above but for the normal distribution having mean 100 and standard deviation 20. Compute a vector of means for all the samples.

2. Display a histogram of the 100 sample means. Compute the mean of the means and compare it to the population mean.

3. Compute the estimated standard error of the mean using the first sample.

4. The standard error you just estimated is estimating the same quantity as the standard deviation of the `means` vector. Compute the latter and compare the two.

5. A medical diagnostic study resulted in the following data. Here a one for disease status indicates disease present, zero indicates disease absent.

   ```
   Test               : 7 -   10 +
   Disease for Test - : 6 -    1 +
   Disease for Test + : 4 -    6 +
   ```

   You can create vectors of raw data representing these results using

   ```
   test ← c(rep('-',7),     rep('+',10))
   dz   ← c(rep('-',6),'+', rep('-',4), rep('+',6))
   ```

   Create and print the `test` and `dz` variables. Estimate the sensitivity of the test, i.e., the conditional probability of a positive test given disease is present. This can be done for example using `mean(test[dz=='+']=='+')`. Also estimate the specificity of the test, i.e., the conditional probability of a negative test given disease is absent. Also get these estimates from `crosstabs(~ test + dz)`.

6. The `lung` dataset is in a directory that is already in the search path so that you can access it without clicking any icon or running any command first. For this dataset, use the `summary` (actually called `summary.formula` internally) function to compute the mean (`summary`'s default statistic) `pat.karno` stratified by `age`, sex, and `wt.loss`, using default interval

construction (quartiles) for continuous variables. Repeat this table to compute instead the mean absolute difference between physician– and patient–estimated Karnofsky status score.

Lab Exercise 4 28 Oct 02

1. An investigator with insufficient funding desires to conduct a clinical study to determine whether a new drug lowers the probability that a patient with a recent stroke will have a recurrence of stroke. The probability of a re-stroke in the population of interest is estimated to be 0.1 if the patient receives standard therapy, and the investigator believes that a worthwhile probability of re-stroke that the new drug could achieve is 0.05. Compute the total number of patients needed to detect a significant difference in proportions of patients who have a recurrence at the $\alpha = 0.05$ level with power equal to 0.85.

2. Use simulation to estimate the exact power that would be achieved for the sample size you just computed. Before doing the calculation, round the total sample size to the nearest integer.

3. For the same probabilities of outcomes of interest to detect, and for a total sample size of 500, compute the proportion of patients that should be randomized to the standard therapy arm such that the power of the test is maximized.

4. Run the power plot examples in the online help for `bpower` in the Hmisc library.

5. For the `hospital` data set perform a χ^2 test of association between the sex of the patient and whether a bacterial culture was done. Do not use Yates' continuity correction. Use the `Statistics .. Compare Samples .. Counts and Proportions` menu.

6. Compute both the linear and Spearman rank correlation coefficients between temperature and white blood count, and P-values for testing for zero correlation. Also compute the linear correlation using the `Statistics .. Data Summaries` menu in S-PLUS.

7. Use the `Statistics .. Compare Samples` menu under S-PLUS to do a Wilcoxon Mann-Whitney two-sample test to compare the median white blood count for patients getting a bacterial culture to those who didn't. Repeat this analysis using programming commands. Finally, repeat this analysis by testing for a nonzero Spearman rank correlation between bacterial culture and white blood count. Compute the median white blood count stratified by culture.

8. Use the graphical techniques learned in class (histograms, box plots, histogram-density plots, etc.) to compare the distributions of temperatures for women and men in the hospital data set.

9. Use graphical methods to address the question of whether or not the duration of hospitalization is affected by whether or not a patient was administered antibiotics.

Lab Exercise 5 4 Nov 02

1. Consider a random sample of system blood pressures containing the observed values 115, 148, 135, 111, 124. Compute the mean, the sample standard deviation, and the estimated standard error of the mean.

2. Test (2-sided) the null hypothesis that this sample was from a population with mean 120 mmHg, assuming the data are from a truly normal population distribution. Do this two ways, using both high- and low-level functions.

3. Compute a 0.99 confidence interval for the unknown population mean, assuming normality, two ways.

4. In a population, measurements have a standard deviation of 10 and mean of 50 and are normally distributed. Compute the sample size necessary to have 0.9 power to reject the null hypothesis that the population mean is 45. Check this by computing the power at that n.[1]

5. For the same setup, compute the sample size necessary to achieve a margin of error of 5 in estimating the population mean from the sample, with 0.95 confidence. In other words, compute n such that the half-width of the 0.95 confidence interval is 5.

6. In a study, 10 of 20 patients treated with chemotherapy were cured. Use two ways to compute an 0.95 confidence interval for the unknown population probability of cure.

7. For the same data, test the null hypothesis that the cure probability in the population is 0.35.

8. The n required to have a margin of error at the 0.95 level in estimating an unknown probability to within a margin of error of δ is $\frac{0.96}{\delta^2}$, when the true probability is near 0.05. Compute n that would "nail down" the probability to within ± 0.05

[1] You may find more information about these methods along with some probability and sample size calculators at `http://biostat.mc.vanderbilt.edu/twiki/bin/view/Main/ClinStat`.