BMI 713: Computational Statistics for Biomedical Sciences

Assignment 4

September 30, 2010 (due October 7)

Loops and Functions in R

1. Let `M` be the following 4×4 matrix: `M = matrix(1:16, ncol=4)`. We would like to compute the sum of all the elements in the matrix.

   (a) Use a double loop to go through each element of the matrix, adding one number at a time.

   (b) Use a single loop, using a single command to sum up each row or column.

   (c) Use a single command to find the sum of the matrix.

2. In the following problems, do not use any standard `R` functions.

   (a) Write a function that takes a single number and returns double that value.

   (b) Write a function which takes two numbers and returns their sum.

   (c) Write a function which takes a vector of values and returns its variance.

   (d) Write a function which takes a single number x and an integer n and returns an array with n copies of x. If the user does not specify n, use a default value of 2.

   (e) Write a function which takes a vector of numbers and determines whether the vector is a *palindrome*. A vector v is a palindrome if v is equal to the reverse of v. The return value of this function should be `TRUE` if the supplied vector is a palindrome and `FALSE` otherwise.

3. (Optional) John H. Conway's *Game of Life*. The *Game of Life* takes place in an infinitely large matrix where entries in the matrix are either dead (have value `0`) or alive (have value `1`). An initial matrix M_0 represents the first generation of some population. To determine whether a cell *survives* to the next generation, its eight neighboring cells are inspected. The following four rules determine whether the cell will be alive or dead in the next generation:

   - Living cells with fewer than 2 living neighbors die due to underpopulation (inability to find a mate, etc.).
   - Living cells with 2 or 3 living neighbors survive.
   - Living cells with 4 or more living neighbors die due to overpopulation.
   - Dead cells with exactly 3 living neighbors are populated with offspring from neighboring cells and become alive.

   An infinite matrix cannot be stored in any computer's memory, so we will have to use a finite one. Use a 10×10 matrix instead. Cells at the edge of the finite matrix will not have eight neighboring cells as required. Treat all of these missing neighbors as dead.

   (a) Write a function `evolve` which takes a population matrix at time t and advances it, using the above rules, to the time $t + 1$.

   (b) Use a loop to run `evolve` for one hundred iterations on an initial (non-empty) matrix of your choosing and print the final result.

Nonparametric test

4. An experiment on learning in animals measures how long it takes a mouse to find its way through a maze. The mean time is 18 seconds for one particular maze. A researcher thinks that a loud noise will cause the mice to complete the maze slower. She measures how long each of 10 mice takes with and without a noise as stimulus. The results are as in Table 1, where the measurements are in seconds.

Table 1: Time used for mice to complete the maze

mouse	1	2	3	4	5	6	7	8	9	10
Without noise	83.7	80.6	101.5	94.6	76.9	83.1	98.5	98.8	91.1	100.3
With noise	83.6	81.1	102.1	95.5	76.5	83.4	99.3	98.6	92.3	100.6

   (a) Calculate the Wilcoxon signed-rank test statistic T.

   (b) What is the expected mean μ_T and standard deviation σ_T of T under the the null hypothesis?

   (c) Perform the Wilcoxon signed-rank test and report the P-value. Is the test significant?

   (d) Perform an appropriate t-test. Do you get the same conclusion based on the t-test and the Wilcoxon signed-rank test?

5. Little et. al. [1] recently studied the discrimination of healthy people from those with Parkinson's disease (PD) based on a range of biomedical voice measurements. Each row in the table corresponds to one of 195 voice recording. Columns of the table correspond to different attributes of the voice recording. The column named *status* indicates the health `status` of the individual (1 – Parkinson's; 0 – healthy). In this assignment, we are mainly interested in the measurement *recurrence period density entropy* (`RPDE`, the 19th column).

   Read the data file *parkinson.data* into R and save it as a data frame.

```
pakinson=read.table("http://archive.ics.uci.edu/ml/machine-learning-databases/
parkinsons/parkinsons.data",sep=",",header=T)
```

   (a) Split the data frame `pakinson` into two data frames according to the value of `status` and save them into two data frames, `p` (`status = 1`) and `h` (`status = 0`) (To save papers, you only need print the command you used to split the data frame; no need to print the data when you submit your homework. Similar below.).

   (b) Extract the measurements `RPDE` from the data frames `p` and `h` and save them into two vectors `v.p` and `v.h`.

   (c) Plot the boxplots of `v.p` and `v.h` in the same plot.

   (d) Perform Wilcoxon rank sum test to test if the median of `v.p` and `v.h` are the same. What is the value of the test statistic? What is the P-value?

   (e) Perform an appropriate t-test to test if the means of `RPDE` of the two groups are the same. Do you get the same conclusion?

6. (Inspired by an example in [2]) Twelve sets of identical twins underwent psychological tests to measure the amount of aggressiveness in each persons's personality. We are interested in comparing the twins to each other to see if first born twin tends to be more aggressive than the other. The results are as in Table 2, the higher score indicates more aggressiveness.

Table 2: Aggressiveness of the twin

first born X_i:	86	71	77	68	91	72	77	91	70	71	88	87
second twin Y_i:	86	77	76	64	96	72	65	90	65	80	81	72

(a) State the null hypothesis and the alternative hypothesis.

(b) If we want to perform a nonparametric test, which one should be used, the Wilcoxon signed-rank test or the Wilcoxon rank sum test?

(c) Perform the test, report the test statistic and P-value. What conclusion do you get?

Parametric vs nonparametric test

7. Comparison of parametric test and nonparametric test.

   (a) Use the following command to generate two vectors `x` and `y`

   ```
   x = rnorm(10,mean = 10, sd =1)
   y = x + rnorm(10,mean=1,sd=1)
   ```

   (b) Perform the paired t-test to test if the difference between x and y (i.e. $x - y$) is less than zero. What is the P-value?

   (c) Perform the Wilcoxon signed-rank test to see if the difference between x and y is less than zero. What is the P-value?

   (d) Repeat the above process 1000 times. How many simulations out of 1000 simulations does the paired t-test give P-value less than 0.05? How about the Wilcoxon signed-rank test? What conclusion can you get from this simulation?

   Hint: you can use
   `t.test(x,y,paired=T,alternative="less")$p.value`
   and
   `wilcox.test(x,y,paired=T,alternative="less")$p.value`
   to get the P-value of the t-test and the Wilcoxon signed-rank test.

References

[1] Little, M.A., McSharry, P.E., Hunter, E.J., Spielman, J. and Ramig, L.O. (2008). Suitability of dysphonia measurements for telemonitoring of Parkinsons disease. In *IEEE Transactions on Biomedical Engineering*.

[2] Kvam, P.H. and Vidakovic, B. (2007). Nonparametric statistics with applications to science and engineering