STA 410/2102, Spring 2004 — Assignment #1

Due at **start** of class on February 13. Worth 15% of the final mark.

Note that this assignment is to be done by each student individually. You may discuss it in general terms with other students, but the work you hand in should be your own.

This assignment concerns hypothesis tests for *circular data*, in which each data point is an angle. For example, some biologists might set up a bird feeder, and then observe the directons that birds leaving the feeder fly. The biologists might be interested in whether the birds tend to leave in some preferred direction or directions, or whether instead the birds are equally likely to fly away in any direction.

To address this question, we would like to test the null hypothesis that a set of n i.i.d. angular observations, $a_1, \ldots, a_n$, are uniformly distributed over the full range from 0 to 2π (we'll suppose that angles are measured in radians, with 0 being north, then proceeding counter-clockwise). There are many possible ways to test this null hypothesis. In this assignment, you will evaluate two methods, checking whether they produce p-values uniformly distributed over $[0, 1]$ when the null hypothesis is true, and comparing how powerful they are for two circumstances in which the null hypothesis is false.

The first method, which we'll call the "sum test", uses the following test statistic:

$$S \;=\; \frac{2}{n}\left[\left(\sum_{i=1}^{n}\cos(a_i)\right)^2 + \left(\sum_{i=1}^{n}\sin(a_i)\right)^2\right]$$

If the null hypothesis is true, and n is large, the joint distribution of the two sums above will, by the Central Limit Theorem, be approximately bivariate normal, with each sum having mean zero and variance $n/2$, and with the two sums being approximately independent. The distribution of the test statistic S will therefore be approximately chi-squared with two degrees of freedom. In some circumstances in which the null hypothesis is false (ie, $a_1, \ldots, a_n$ are not uniformly distributed over 0 to 2π) S might usually be larger than is typical under the null hypothesis. We might therefore reject the null hypothesis if the observed value of S is large. We can quantify the strength of the evidence agains the null hypothesis by means of a p-value equal to the probability that a value from a chi-squared distribution with two degrees of freedom will be as large or larger than the observed value of S. We can compute this p-value using the `pchisq` function in R.

A quite different approach, which we'll call the "gap test", is to look at the largest gap between data points, ordering them around the circle. If $a_{(1)}, \ldots, a_{(n)}$ are the data points arranged in increasing order, then the largest gap will be

$$G \;=\; \max\left(a_{(2)} - a_{(1)},\; \ldots,\; a_{(n)} - a_{(n-1)},\; a_{(1)} + 2\pi - a_{(n)}\right)$$

If the data is not uniform, we would expect to have a larger value of G than if the data is uniform, since a non-uniform distribution will have some regions with lower probability density. We therefore define the p-value for this test to be the probability that a value for G as large or larger than its value for the actual data would be produced if the data were actually uniformly distributed. There may be a way to find this easily, with some theoretical work, but let's suppose that we're not that clever. We can still compute the p-value by simulating many data sets in which the points are uniformly distributed, and comparing the observed value of G to the values of G for these data sets.

You should write R functions called `sum.test` and `gap.test`, which implement these two hypothesis tests. Both procedures should take a vector of data points as an argument, and return the p-value for the hypothesis test as their value. The `gap.test` function should take the number of data sets to simulate in order to compute the p-value as an additional argument, which should default to 99. You should use this default value for this assignment, even though it is smaller than would be advisable in practice, since otherwise the tests below might take a long time.

You should also write a function called `sim.unif`, which takes as its first argument a function for computing the p-value of a test of the null hypothesis that the data is uniformly distributed. This function should simulate the distributions of p-values produced by this test when the null hypothesis is true. Its second argument should be the size of data set to simulate (from the uniform distribution over 0 to 2π). Its third argument should be the number of data sets to simulate. The value of `sim.unif.test` should be a vector of p-values, one for each simulated data set.

You should use `sim.unif` to produce histograms of the distribution of p-values for the sum test and the gap test, for data sets of size 2, 4, 10, and 20. You should simulate at least 20000 data sets of each size for the sum test, but to save time, you may simulate only 1000 data sets for the gap test. You should discuss the results, noting any circumstances in which either of these tests should not be used.

You should also write a function called `sim.nonunif`, which also takes as its first argument a function for computing the p-value of a test of the null hypothesis that the data is uniformly distributed. This function should simulate the distributions of p-values produced by this test when the null hypothesis is *false*. The second and third arguments of this function should again be the size of the data sets to simulate and the number of data sets to simulate. This function should take additional arguments, however, which specify what sort of non-uniform distribution should be used for the data sets.

Here is how you should produce non-uniform distributions. You should start by generating angles uniformly from 0 to 2π, which you can convert to points, (x, y), on the unit circle. You can then modify these points by either adding a constant to x or by multiplying x by a constant, producing a new point, (x', y). Finally, you can convert back to angles by finding the angle that (x', y) makes with the x axis. This can be done using the `atan2` function. The amount to add to x and the amount to multiply x by should be additional arguments of `sim.nonunif`.

You should use `sim.nonunif` to produce histograms of the distribution of p-values for the sum test and the gap test for data sets of size 2, 4, 10, and 20, in which the data is non-uniformly distributed according to the distribution produced by adding 0.5 to x, or by multiplying x by 1.7. Again, you should simulate at least 20000 data sets for the sum test, but you may simulate only 1000 data sets for the gap test. Note that these tests may take up to 10 minutes or more of computer time, so you will want to debug your program using a smaller number of data sets. You should discuss the results, noting in what circumstances each test has reasonable power (ie, when it has a good chance of rejecting the null hypothesis at some usual level, such as 0.05 or 0.1).

You should organize your program and your experiments in a good fashion, which would allow someone else to understand what you did, and reproduce your results. You *must* indent your program properly, in a *consistent* way. You should include comments describing the way the functions should be used, and clarifying any parts of the program that might be puzzling. You should hand in your program, the plots produced by your program, and your discussions of the results.

You may find the R `sort` function to be useful. The `prob`, `nclass`, `xlim`, and `xlab` options to the histogram function, `hist`, may also be useful.