ESAM 448 Assignment Collaboration Policy

Overview

A large portion of the learning objectives in this class are associated with the computer assignments. Explanations and demonstrations are given in class, of course, but **nothing can replace the hands-on experience gained by actually implementing these algorithms yourself.**

Individual work

Assignments in this class are intended to be primarily individual efforts. Northwestern's Basic Standards of Academic Integrity state that you may not submit "material that in part or whole is not entirely one's own work without attributing those same portions to their correct source." Here in this class, this means that you may discuss *general* approaches to the assigned problems with other students but **your code and your write-up must be your own.**. **You may not use materials produced as course work by other students, whether in this term or previous terms, nor may you provide work for other students to use.**

It's good to help other students, and discussing problems assists learning. As a general rule, however, during any time that you are discussing problems with another student, **your own solution should not be visible**, either to you or to them. Make a habit of closing your laptop while you're helping.

Submitted programs may be checked for copying.

Attributing other sources

Writing code is similar to academic writing in that **when you use or adapt code from another source as part of an assignment, you must cite that source**. This should include both comments in any written assignment and an inline comment in your submitted code. These comments not only ensure you are giving proper credit, but help with code understanding and debugging during grading. Whether you are copying a snippet of code or an entire module, you should credit the source.

You may also use material from external sources, as long as: (1) the material is available, in principle, to all students in the class (i.e., it should be *publicly* available; it is not necessary for you to let other students know about the source, however); (2) you give proper attribution; and (3) the assignment itself allows it. **In particular, if the assignment says "implement X," then you must create your own X, not reuse someone else's.**

It's acceptable to use any code provided in class or on the course web page without need for attribution, however. You may also re-use designs, ideas and code from your own work earlier in the quarter.

General Matlab advice

Some of the simulations you will do will require a large number of samples. To make the running of your programs as efficient as possible (i.e., to keep the running times as small as possible), you should make every effort to *vectorize* your programs. A full discussion would take more space than is available here, but one bit of advice is to minimize the use of loops. See `https://www.mathworks.com/help/matlab/matlab_prog/vectorization.html` for a longer discussion.

Assignment instructions: Completing this assignment requires submitting, for each problem, a written part or a `Matlab` program, and in some cases, both of these. Any written part should be handed in during class; Matlab programs should be emailed to `esam448@u.northwestern.edu`. Please comment your code to explain your implementation. Including *all* of the programs for a given assignment in a single email will be appreciated.

1. A derangement of the numbers 1 through N is a permutation of all N of those numbers such that none of them is in the "right place." For example, 34251 is a derangement of 1 through 5, but 24351 is not because 3 is in the 3rd position. Write a Matlab function (using the specific function name `lastname_firstname_hw1_prob1`)

```
function [prob_est,stderr_of_est] = lastname_firstname_hw1_prob1(M,N)

[...any statements you need...]

end
```

   that generates M random permutations of the integers $1, 2, \ldots, N$ and then determines how many of those permutations is a derangement. The function should return an estimate for the probability of a derangement occurring, i.e., $P\{$ a permutation of $1,\ldots,N$ is a derangement $\}$ *and* an estimate for the standard error associated with that estimate.

   Use your Matlab function to determine the estimated probability when $N = 12$ and report the number of samples necessary to reduce the standard error in that estimate to 0.0001.

2. Consider the following game: you begin with \$K. You flip a coin — that may or may not be fair — winning \$1 if the coin lands heads and losing \$1 if the coin lands tails. The coin flips continue until you either go broke or have \$100. Write a Matlab function that takes the starting value `K`, a number of samples `M`, and a probability `p` of heads

```
function [prob_est,stderr_of_est] = lastname_firstname_hw1_prob2(M,K,p)

[...any statements you need...]

end
```

   that estimates the probability of reaching \$100 and the estimate's standard error from `M` samples. (Here, one sample is considered to be one sequence of coin flips that reaches either \$100 or \$0.)

   First, check your program for the case `p=0.5`; by symmetry the probability of reaching \$100 should be 0.5 in this case. Then determine the probability of reaching \$100 if the probability of the coin producing heads on a single flip is reduced to `p=0.49`.

3. Consider picking three points in the (x, y) plane where the x and y coordinates of each point are independently chosen from a standard normal distribution. These three points, of course, determine a triangle. Write a Matlab function

```
function [prob_est,stderr_of_est] = lastname_firstname_hw1_prob3(M)

[...any statements you need...]

end
```

that uses M independently sampled triangles to determine the probability that such a randomly chosen triangle is obtuse. Also return the standard error of the estimate.

User your Matlab function to estimate this probability. Can you make a guess as to what the right answer should be? Do you think it's reasonable to try to justify this guess by increasing the number of samples?

4. Use a change of variable with an appropriate transformation and Matlab's uniform `rand` function to generate random values from an exponential probability distribution, $p(x) = ae^{-ax}$, $x \geq 0$. Your write-up should give the details explaining how this works. Then for $a = 1$ generate $N = 1000$ samples and make a plot [include this in your written solutions] comparing the sample cumulative distribution function (`Matlab` function `ecdf`) with the exact answer. Next, write a Matlab function with the structure

```
function [chi2pvals] = lastname_firstname_hw1_prob4(M,N,a,nbins)
chi2pvals=zeros(M,1)

[...other statements you need...]

end
```

This function should generate a vector of N exponential random variables with parameter a multiple (M) times. For each N-vector of samples, it should use the χ^2 test (`Matlab`: `chi2gof`) to compare the generated data with the theoretical distribution. For the χ^2 test, divide the full range from 0 to ∞ into `nbins` in such a way so that the expected number of samples falling in each bin will be equal (note: these will not be equal width bins). The function should return a vector giving the M p-values from the χ^2 tests. If `M=1000`, `N=1000`, `a=1` and `nbins=10`, how many times does the vector of samples pass the χ^2 test at a 0.95 significance? For these parameters, include in your written solution a plot showing a histogram of the p-values using 20 equal width bins. What does the distribution of p-values look like? Can you explain this? Finally, include your function in the email to `esam448@u.northwestern.edu`.