CS 533 – Experimental Methods in Computer Science

Spring 2025

Project #3: Time Performance Analysis (120 pts)

Goal: In this project you will apply the methods that you have learned to assess time performance of an algorithm. The program you will use to evaluate time performance is the program you wrote in Project 2, namely the Hopkins statistic code.

Hints:

- You may use any profiler that is appropriate for the language in which you wrote your code. For instance, in Matlab you may use the "Run and Time" option to open the profiler, and also the functions "tic", "toc", "timeit", and "cputime".
- Please report the machine specifications when reporting time performance. Additionally, report the version of the software you are running.

Questions:

Based on what we discussed in class, answer the following questions for the Hopkins statistic estimation program that you wrote in the previous project:

- Before running your code, answer:

(1) What are the main functions your algorithm calls (These could be system functions or user functions)? What kind of operations do these functions perform? Which of these operations is the dominant cost of your algorithm? (20 pts)

(2) Without running your program and based on your answers from (1), what would you expect the time complexity of your algorithm to be? Does it depend on the number of initial data points *n*, or on the number of random points generated *m*, or on the number of bootstrapping steps *B*, or on a combination of these, or on any other factor? (20 pts)

(3) Design experiments that can test your answers to (1) and (2). Describe these experiments in detail before you run them. (Hint: What is the question you're trying to answer? Which factors can you vary? What is your input and expected output? Etc.) (20 pts)

- Run your experiments and answer:

(4) What is the output of the profiler? Which functions are called the most, and which functions have the highest time cost? Do the number of calls to a function and time spent in performing that function correlate? Do your observations agree with your answers from (1)? (20 pts)

(5) Plot the results of your experiments that test your answer to (2). Does the measured time complexity match what you expected from analyzing your algorithm? (20 pts)

(6) Do you observe any change in time performance for experiments performed under light system load as opposed to heavy system load? Does CPU time match wall time in either case? Report the observed values. (20 pts)

Reference:

[1] McGeoch, Catherine C. *A guide to experimental algorithmics*. Cambridge University Press, 2012. https://libproxy.unm.edu/login?url=https://search.ebscohost.com/login.aspx?authtype=ip,guest&custid=s4858255&groupid=main&profile=eds&direct=true&db=nlebk&AN=416746&site=eds-live&scope=site