



**HDS 5230: High Performance Computing**

**Assignment 3**

**Harika Pamulapati**

**Professor: Adam Doyle**

**1. Tabulate the execution times of each of the individual approaches for computing distance in Python (i.e., run the shared code on your computer, note the times, and tabulate them).**

Execution time using for loop – 0.011 seconds

Execution time using apply function – 0.006 seconds

Execution time using Vectorization – 0.001 seconds

**2. Next, replicate the for-loop based approach (the first one) and two different ways to make that version more efficient, in R. Profile these three approaches, and tabulate the results.**

Executing time using for loop – 0.01 seconds

Execution time using vectorized using sapply – 0.04 seconds

Execution time using fully vectorized – 0 seconds

**3. Based on the computational efficiency of implementations in Python and R, which one would you prefer? Based on a consideration of implementation (i.e., designing and implementing the code), which approach would you prefer? Taking both of these (run time and coding time), which approach would you prefer?**

The vectorized method in Python achieves execution times of 0.001 seconds while R operates at 0 seconds thus showing similar efficiency levels for both programming languages. The syntax of R vectorization provides natural implementation benefits because its design as a vectorized operation system simplifies writing vectorized code without requiring explicit vectorization thinking. The integrated evaluation of execution speed and development time indicates that I would choose Python over R given that performance variations become irrelevant for typical

applications but Python provides continuous familiar programming structure for various tasks. This distance calculation feature fits naturally into complex systems because it works across various programming components including web services and machine learning tools.

**4. Identify and describe one or two other considerations, in addition to these two, in determining which of the two environments – Python or R – is preferable to you.**

The choice between Python and R depends mainly on two essential factors: test automation capabilities in addition to staff proficiency levels. The extensive Python ecosystem includes features beyond data analysis which makes it more advantageous than R for projects needing system interface capabilities and general programming tasks. Python offers broader language availability because it serves as a standard introductory programming language while R maintains enthusiast programmers mainly in statistics and analytics. The main selection factors between these languages will usually surpass minor differences in implementation together with execution time duration.