1 Standards Assessed

I believe this project helps meet all of its goals. For libraries and linking, it forces students to understand how to work in two different para-dimes, using skills from both sections of this course to create a final project.

For error handling, this comes from the custom exception throwing that must be done when taking in the data for the function, as care has to be taken to ensure that there are no domain flaws and that the data types align. This focus on data types is especially insightful, as you transfer the code from a laissez faire language to a quite strict one.

Testing is, intuitively, in the test file the students are required to create. This project also introduces one way to test, by timing functions versus each other. Functions, also intuitively, come from the fact that each of the coded equations is a function.

Finally, this project provides a nice “under the hood” for the standard library, showing how these functions are actually calculated and allowing students to understand their limitations.

2 Exam Questions

This question has 4 parts. Please give a complete answer to each part. You are a software engineer at IBM working on a new data science tool and, to your dismay, the Python package you’re writing goes slowly, even when using Numpy. You decide that the best path forward is to write some code in trusty ol’ C++, and then call those functions in Python. For parts (a)-(b), refer to equation (1).

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2} \tag{1}$$

where μ is the mean and N is the number of elements.

(a) Write a C++ function “double mean(const std::vector<Double> &input)” which calculates the mean. You may use standard vector functions.

(b) Write a C++ function “double std_dev(const std::vector<double> &input)” which calculates the standard deviation. You may use the C++ math library to call pow(x, y). (Hint: how can you use powers to calculate square roots?)

(c) Explain, but do not write out, how to make a C++ function which takes a Python array of doubles as an input and converts them into a vector of doubles.

(d) Write PyObject *Mean(PyObject *self, PyObject* args), the C++ function which takes in a Python array of doubles in args, converts them to a vector of doubles, and returns the standard deviation, calling std_dev().