Task Splitting

Listed below is a list of the different tasks from the Project and who was responsible for each task, and what they did.

1. CHPC instructions, done by Mike
2. Implementations
   1. Serial

Framework by Mike, completed by Nathan. Mike laid out a solid initial structure for the serial KMeans implementation. Nathan took over to resolve remaining bugs, finalize the algorithm, and ensure it was robust enough to serve as a baseline for comparison with the parallel versions.

* 1. Parallel Shared Memory CPU

Implemented by Mike. Straightforward to implement off of the back of the serial implementation.

* 1. Parallel CUDA GPU

Implemented by Mike. Some refactoring was necessary here to transfer all data structures away from vectors (which are not supported by kernel functions).

* 1. Distributed Memory CPU

Implemented by Nathan. Extended the serial KMeans algorithm to run in a distributed memory environment using MPI. He handled data partitioning across processes and implemented inter-process communication for parallel clustering.

* 1. Distributed Memory GPU

Framework by Nathan, completed by Mike. Nathan set up the initial structure by adapting the MPI-based CPU distribution approach for use with GPUs. Mike then finalized the implementation by enabling GPU-specific data broadcasting and processing across MPI ranks. He also handled all of the CHPC integration.

1. Scaling Studies, done by Mike.
2. Validation function, framework done by Nathan, completed by Mike.
3. Reuse code across implementations, done by both.
4. Visualization of the output: Done by Nathan in Python.
5. Responsibility text, Done by both.