

## Calculating K-Nearest Neighbors using OpenMP

Team Member: Yubing Hou, Nema Nemati, August Meyer

This project is about using OpenMP to calculate the K-Nearest Neighbors (KNN), a simple non-parametric method used for classification and regression in machine learning and statistics. The algorithm would be naive: for any given vector  $v_i$ , the algorithm will calculate the distance between this vector with all other vectors, and find out the k-nearest vectors. For determine the distance between two vectors  $a$  and  $b$ , we use this equation:

$$d = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + \dots + (a_i - b_i)^2}$$

The HPC technique would be used in this project would include serial optimization, cache optimization, and multithreading. Serial optimization and cache optimization can be used for calculating the distance between two vectors. Multithreading would be used to search for the k-nearest vectors in parallel.

Development Plan:

### Stage 1: Data Generation

In this stage, we design and implement program that can randomly generate data for testing. Data can be generated sequentially. We would need a program that can generate certain amount of vectors with a specified dimension. Data output would be saved in a text file for the main program to read.

### Stage 2: Implement KNN Serial Code

In this stage, we would design and implement the naive and serial KNN code using C. This program should be able to read the data from the data generator program and store them in array, and the naive algorithm should allow us to find the k-nearest neighbor of a specified data point.

### Stage 3: Code Optimization

In this stage, we add OpenMP code that make certain loop parallel. We would also looking into the iteration of loops to improve temporal and spatial locality. Finally, we would add parallelism in the searching part to find out KNN faster.

### Stage 4: Verification and Debugging

We will compare our results with the output using other packages to verify the correctness of our code. If anything is not matching, we will debug and make sure everything works correctly.