CMSC 691 – High Performance Distributed Systems

Virginia Commonwealth University, Fall 2016

Due date: September 16, 2016

Big data mining involves the use of datasets with millions of instances. The computational complexity of machine learning methods limits their scalability to larger datasets. The simplest classifier is the nearest-neighbor classifier (1NN) but its computational complexity is O(n^2), where n is the number of instances.

You are provided the code for a sequential 1NN algorithm, which computes for each data instance the distances to the other instances, and predicts the data class using the instance located within the smallest Euclidean distance. Then, the accuracy of the classifier is measured as the relation between the number of successful predictions and the number of instances. You may run the code using the datasets provided and analyze how the runtime increases with regards of the size of the dataset.

Your job is to parallelize the code using threads and conduct all the code optimizations you consider relevant to speed up its execution, as long as the output is correct. Should you consider critical sections in your code, be aware of data races and employ the tools to guarantee the mutual exclusion. The faster solution the better grade you get!

1. Compare the runtime of the serial and parallel versions considering the datasets with different sizes. Calculate the speedup using 2, 4, and 8 threads.
2. Estimate the proportion of parallel code by relating the speedup you get and the number of threads employed. What's the maximum speedup you would be able to obtain using an infinite number of threads and cores?
3. Let's run using 256 threads. What's the speedup now?